FORM F.SK05
BUKTI BIMBINGAN SKRIPSI

Nama Mahasiswa : Nanda Mei Sitasari

NIM : 202400010

Judul Skripsi : Faktor-Faktor yang Mempengaruhi Persepsi Masyarakat Surabaya terhadap Surat Edaran Menteri Sosial No. 2 Tahun 2023 tentang Pengemis Online

Dosen Pembimbing : Alfisyahrina Hapsery, S.Si., M.Si.

Materi Pembimbingan Skripsi	Tanda Tangan Dosen Pembimbing
1. Bimbingan Skripsi dari Bab 1-5	
2. Perbaikan terkait Interpretasi untuk Statistka deskriptif dan format skripsi	
3. Perbaikan dalam pengkategorian data	
4. Perbaikan untuk variabel pendidikan untuk dijadikan 2 kategori	
5. Penambahan nilai Relative Risk (RR)	
6. Perbaikan untuk kesimpulan	
7. Ada perubahan terkait Judul	
8. Penambahan karakteristik responden	

PROGRAM STUDI STATISTIKA
SK BAN-PT No. 1765/SK/BAN-PT/AK-PPJ/S/III/2022
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS PGRI ADI BUANA SURABAYA

FORM F.SK08

PERBAIKAN/REVISI SEMINAR DAN UJIAN SKRIPSI

Nama Mahasiswa : Nanda Mei Sitasari

NIM : 202400010

Judul Skripsi : Faktor-Faktor yang Mempengaruhi Persepsi Masyarakat Surabaya terhadap Surat Edaran Menteri Sosial No. 2 Tahun 2023 tentang Pengemis Online

Dosen Pembimbing : Alfisyahrina Hapsery, S.Si., M.Si.

Materi Revisi Seminar dan Ujian Skripsi	Tanda Tangan Dosen Penguji
1. Penambahan penjelasan untuk pengkategorian data	
2. Penambahan indikator yang sudah reliabel	
3. Penjelasan untuk screening kuesioner lebih diperjelas	
4. Penambahan sumber yang mendukung perihal pengkategorian data	
5.	
6.	

Surabaya, 10 Juli 2024

Dosen Pembimbing,

Alfisyahrina Hapsery

NPP. 1804856/DY

Lampiran 1 Kuesioner

Screening

Apakah Anda sudah bekerja atau berusia 18 tahun ke atas dan memiliki akun TikTok (pernah menyaksikan konten pengemis *online*)?

a. Iya
b. Tidak

Catatan: Jika responden tidak mengetahui Surat Edaran Menteri Sosial No. 2 Tahun 2023, maka informasi mengenai surat edaran tersebut akan disampaikan.

Petunjuk Umum

Anda dimohon untuk memberikan tanggapan yang sesuai atas pernyaan berikut, Anda diminta untuk memilih salah satu alternatif pilihan yang tersedia berdasarkan keadaan Anda yang sesungguhnya dengan cara memberi tanda centang (√) pada salah satu pilihan yang paling sesuai

A. Identitas Responden

Isilah data diri anda sesuai dengan keadaan yang sebenarnya pada kolom jawaban yang telah disediakan.

1. Nama Lengkap :
2. Jenis Kelamin : □ Laki-laki □ Perempuan
3. Usia :
4. Domisili saat ini :
5. Pekerjaan :
6. Pendidikan Terakhir : □ SD □ SMP □ SMA □ D3/S1

B. Sudut pandang masyarakat tentang setuju atau tidak setuju nya terhadap isi dari Surat Edaran Menteri Sosial No. 2 Tahun 2023 untuk menindak maraknya fenomena pengemis *online* lansia di media sosial Tiktok

1 : Sangat Tidak Setuju
2 : Tidak Setuju
3 : Setuju
4 : Sangat Setuju

No.	Indikator	1	2	3	4
1.	Para lanjut usia, anak, penyandang disabilitas dan atau kelompok rentan lainnya yang menjadi korban eksploitasi melalui mengemis online berhak mendapatkan perlindungan				
2.	Mengemis secara online sangat meresahkan masyarakat				
3.	Mengemis secara online dapat menurunkan kualitas masyarakat				
4.	Memberikan bantuan sosial terhadap korban pengemis online sebagai bentuk upaya pemerintah memberantas pengemis online				
5.	Masyarakat wajib melapor ke pihak berwajib jika menemui adanya kegiatan mengemis secara online				
6.	Diberikan sanksi tegas bagi pelanggar kebijakan				
7.	Mengemis secara online sangat mengganggu ketertiban umum				
8.	Pengemis memiliki hak untuk memanfaatkan perkembangan IPTEK				
9.	Pengemis perlu berkreasi untuk mendapatkan belas kasihan masyarakat				
10.	Konten mengemis secara online perlu diblokir				

C. Pemahaman dan pengetahuan yang dimiliki oeh masyarakat terhadap Surat Edaran Menteri Sosial No. 2 Tahun 2023 tentang untuk menindak maraknya fenomena pengemis *online* lansia di media sosial Tiktok

1 : Tidak Tahu
2 : Kurang Tahu

3 : Tahu
4 : Sangat Tahu

No.	Indikator	1	2	3	4
1.	Mengetahui isi surat edaran Menteri Sosial no. 2 Tahun 2023 tentang penertiban kegiatan eksploitasi dan/ kegiatan mengemis secara online				
2.	Mengetahui bahwa para lanjut usia, anak, penyandang disabilitas dan atau kelompok rentan lainnya menjadi korban atau pelaku utama dari mengemis online.				
3.	Mengetahui maksud dan tujuan dari surat edaran Menteri Sosial no. 2 Tahun 2023				
4.	Mengetahui upaya apa saja yang dilakukan pemerintah untuk memerangi pengemis online				
5.	Mengetahui sanksi yang diberikan kepada pelanggar kebijakan				

D. Perilaku masyarakat terhadap kinerja pemerintah yang diwujudkan dengan tindakan-tindakan terhadap pencapaian hasil dan harapan masyarakat.

1 : Sangat Tidak Setuju
2 : Tidak Setuju
3 : Setuju
4 : Sangat Setuju

No.	Indikator	1	2	3	4
1.	Para lanjut usia, anak, penyandang disabilitas dan atau kelompok rentan lainnya yang menjadi				

	korban kegiatan mengemis online harus di perhatikan				
2.	Upaya yang dilakukan oleh pemerintah harus di tinjau kembali				
3.	Mendukung upaya pemerintah untuk memerangi masyarakat yang mengemis secara online				
4.	Bantuan sosial dan rehabiitasi tidak efektif untuk mengurangi pengemis online				
5.	Membantu mendukung upaya pemerintah dengan tidak menyawer pengemis online				
6.	Sanksi yang diberlakukan dinilai tidak berpengaruh atau tidak tegas				

E. Religiusitas.

1 : Sangat Tidak Setuju
2 : Tidak Setuju
3 : Setuju
4 : Sangat Setuju

No.	**Indikator**	**1**	**2**	**3**	**4**
1.	Saya lahir dikeluarga yang paham dan taat agama				
2.	Saya menempuh pendidikan di sekolah berbasis agama				
3.	Saya termasuk orang yang rajin bersedekah				
4.	Saya lebih suka bersedekah kepada anak yatim daripada bersedekah kepada pengemis				

5.	Saya lebih suka bersedekah kepada penjual koran daripada bersedekah kepada pengemis				
6.	Saya lebih suka bersedekah kepada pedagang yang jualannya tidak laku daripada bersedekah kepada pengemis				

F. Faktor Ekonomi.

1 : Sangat Tidak Setuju
2 : Tidak Setuju
3 : Setuju
4 : Sangat Setuju

No.	**Indikator**	**1**	**2**	**3**	**4**
1.	Saya bisa menabung setiap bulan sebesar 30% dari total gaji				
2.	Saya bisa memenuhi kebutuhan pokok setiap bulan sebesar 50% dari total gaji				
3.	Saya bisa memenuhi keinginan pribadi setiap bulan sebesar 20% dari total gaji				
4.	Saya tidak kesulitan dalam membeli apapun				
5.	Saya tidak memiliki pinjaman uang				

G. Gaya Hidup.

1 : Sangat Tidak Setuju
2 : Tidak Setuju
3 : Setuju
4 : Sangat Setuju

No.	**Indikator**	**1**	**2**	**3**	**4**

1.	Saya senanga berbelanja online di Tiktok				
2.	Saya merasa tertarik untuk beerbelanja ketika ada barang unik				
3.	Saya suka membeli gift di Tiktok				
4.	Saya terkadang memberikan gift kepada pengguna yang live Tiktok				
5.	Saya merasa tertarik untuk berbelanja ketika ada promo di live Tiktok				

H. Intensitas Bermain Gadget.

1 : Sangat Tidak Setuju
2 : Tidak Setuju
3 : Setuju
4 : Sangat Setuju

No.	**Indikator**	**1**	**2**	**3**	**4**
1.	Saya menghabiskan waktu untuk membuka Tiktok lebih dari 30 menit dalam sehari				
2.	Saya selalu membawa gadget kemanapun				
3.	Saya selalu menggunakan internet setiap hari				
4.	Saya merasa gelisah jika dalam sehari tidak menggunakan gadget				
5.	Saya tidak senang jika dalam sehari tidak bermain Tiktok				
6.	Saya menggunakan *e-money* atau *mobile banking*				
7.	Saya senang melakukan live di Tiktok				

8.	Saya senang membuat video pendek atau konten di Tiktok				

Lampiran 2 Surat Edaran Menteri Sosial No. 2 Tahun 2023

SALINAN

MENTERI SOSIAL REPUBLIK INDONESIA

Yth:
1. Para gubernur; dan
2. Para bupati/wali kota,

di – tempat

SURAT EDARAN
NOMOR 2 TAHUN 2023
TENTANG
PENERTIBAN KEGIATAN EKSPLOITASI DAN/ATAU KEGIATAN MENGEMIS YANG MEMANFAATKAN LANJUT USIA, ANAK, PENYANDANG DISABILITAS, DAN/ATAU KELOMPOK RENTAN LAINNYA

A. Latar Belakang

Bahwa sesuai dengan ketentuan Kitab Undang-Undang Hukum Pidana, Undang-Undang Nomor 13 Tahun 1998 tentang Kesejahteraan Lanjut Usia, Undang-Undang Nomor 23 Tahun 2002 tentang Perlindungan Anak, Undang-Undang Nomor 21 Tahun 2007 tentang Pemberantasan Tindak Pidana Perdagangan Orang, dan Peraturan Pemerintah Nomor 31 Tahun 1980 tentang Penanggulangan Gelandangan dan Pengemis, dengan melakukan eksploitasi dan/atau kegiatan mengemis menyebabkan keresahan masyarakat dan mengganggu ketertiban umum, sehingga perlu menerbitkan Surat Edaran tentang Penertiban Kegiatan Eksploitasi dan/atau Kegiatan Mengemis yang Memanfaatkan Lanjut Usia, Anak, Penyandang Disabilitas, dan/atau Kelompok Rentan Lainnya.

B. Maksud dan Tujuan

1. Mencegah adanya kegiatan mengemis baik secara *offline* dan/atau *online* di media sosial yang mengeksploitasi para lanjut usia, anak, penyandang disabilitas, dan/atau kelompok rentan lainnya.
2. Melindungi para lanjut usia, anak, penyandang disabilitas, dan kelompok rentan lainnya dari eksploitasi yang dilakukan dengan kegiatan mengemis secara *offline* dan/atau *online* di media sosial.

C. Ruang Lingkup

Surat Edaran ini memuat himbauan kepada gubernur dan bupati/wali kota di seluruh Indonesia untuk mencegah dan menindak kegiatan mengemis baik yang dilakukan secara *offline* maupun *online* di media sosial yang mengeksploitasi para lanjut usia, anak, penyandang disabilitas, dan/atau kelompok rentan lainnya serta memberikan perlindungan kepada para lanjut usia, anak, penyandang disabilitas, dan/atau kelompok rentan lainnya yang menjadi korban eksploitasi dari kegiatan mengemis baik dilakukan secara *offline* dan/atau *online* di media sosial.

D. Dasar Hukum

1. Kitab Undang-Undang Hukum Pidana;
2. Undang-Undang Nomor 13 Tahun 1998 tentang Kesejahteraan Lanjut Usia (Lembaran Negara Republik Indonesia Tahun 1998 Nomor 190, Tambahan Lembaran Negara Republik Indonesia Nomor 3796);
3. Undang-Undang Nomor 23 Tahun 2002 tentang Perlindungan Anak (Lembaran Negara Republik Indonesia Tahun 2002 Nomor 109, Tambahan Lembaran Negara Republik Indonesia Nomor 4235) sebagaimana telah diubah dengan Undang-Undang Nomor 35 Tahun 2014 tentang Perubahan atas Undang-Undang Nomor 23 Tahun 2002 tentang Perlindungan Anak (Lembaran Negara Republik Indonesia Tahun 2014 Nomor 297, Tambahan Lembaran Berita Negara Republik Indonesia Nomor 5606);
4. Undang-Undang Nomor 21 Tahun 2007 tentang Pemberantasan Tindak Pidana Perdagagangan Orang (Lembaran Negara Republik Indonesia Tahun 2007 Nomor 58, Tambahan Lembaran Negara Republik Indonesia Nomor 4720);
5. Undang-Undang Nomor 11 Tahun 2008 tentang Informasi dan Transaksi Elektronik (Lembaran Negara Republik Indonesia Tahun 2008 Nomor 58, Tambahan Lembaran Negara Republik Indonesia Nomor 4967) sebagaimana telah diubah dengan Undang-Undang Nomor 19 Tahun 2016 tentang Perubahan Atas Undang-Undang Nomor 11 Tahun 2008 tentang Informasi dan Transaksi Elektronik (Lembaran Negara Republik Indonesia Tahun 2016 Nomor 251, Tambahan Lembaran Negara Republik Indonesia Nomor 592);
6. Undang-Undang Nomor 11 Tahun 2009 tentang Kesejahteraan Sosial (Lembaran Negara Republik Indonesia Tahun 2009 Nomor 12, Tambahan Lembaran Negara Republik Indonesia Nomor 4967);
7. Undang-Undang Nomor 8 Tahun 2016 tentang Penyandang Disabilitas (Lembaran Negara Republik Indonesia Tahun 2016 Nomor 69, Tambahan Lembaran Negara Republik Indonesia Nomor 5871); dan
8. Peraturan Pemerintah Nomor 31 Tahun 1980 tentang Penanggulangan Gelandangan dan Pengemis (Lembaran Negara Republik Indonesia Tahun 1980 Nomor 51, Tambahan Lembaran Negara Republik Indonesia Nomor 3177).

E. Isi Surat Edaran

Para gubernur dan bupati/wali kota di seluruh Indonesia, dihimbau untuk:

1. mencegah adanya kegiatan mengemis baik yang dilakukan secara *offline* maupun *online* di media sosial yang mengeksploitasi para lanjut usia, anak, penyandang disabilitas, dan/atau kelompok rentan lainnya;
2. apabila ditemukan kegiatan mengemis dan/atau yang mengeksploitasi para lanjut usia, anak, penyandang disabilitas, dan/atau kelompok rentan lainnya harus melaporkan kepada Kepolisian Negara Republik Indonesia dan/atau ditindaklanjuti melalui Satuan Polisi Pamong Praja; dan
3. memberikan perlindungan, rehabilitasi sosial, dan bantuan kepada para lanjut usia, anak, penyandang disabilitas, dan/atau kelompok rentan lainnya yang telah menjadi korban eksploitasi melalui

mengemis baik yang dilakukan secara *offline* maupun *online* di media sosial.

F. Penutup

Demikian Surat Edaran ini dibuat untuk diketahui dan dilaksanakan sebagaimana mestinya.

Ditetapkan di Jakarta
pada tanggal 16 Januari 2023

MENTERI SOSIAL REPUBLIK INDONESIA,

ttd

TRI RISMAHARINI

Salinan Surat Edaran ini disampaikan kepada Yth:
1. Presiden Republik Indonesia.
2. Wakil Presiden Republik Indonesia.
3. Menteri Bidang Koordinator Bidang Pembangunan Manusia dan Kebudayaan.
4. Menteri Dalam Negeri.
5. Kepala Kepolisian Negara Republik Indonesia.
6. Menteri Komunikasi dan Informasi.

Salinan sesuai dengan aslinya
Plt. Kepala Biro Hukum

Evy Flamboyan Minanda
198102182006042002

Lampiran 3 Data Penelitian

X1	X2	X3	X4	X5	X7	X8	X9	Y
1	23	0	0	1	0	0	0	0
0	20	1	0	2	1	0	1	1
0	22	0	1	1	0	1	0	0
1	23	1	1	2	1	1	1	1
0	27	0	0	2	0	0	0	1
1	28	0	1	2	0	1	0	1
1	27	0	0	2	0	0	1	0
1	29	1	1	2	1	1	1	0
1	27	0	0	2	1	0	0	1
1	28	0	1	2	1	0	1	0
1	48	0	0	2	0	1	1	1
0	26	0	0	2	1	1	1	0
0	27	1	0	2	1	1	1	1
1	29	0	0	2	0	0	0	0
1	25	0	1	1	0	0	0	1
0	28	1	1	2	0	1	1	1
1	25	0	0	2	1	0	0	1
0	30	0	1	2	0	0	0	0
0	24	1	0	1	1	1	0	0
1	24	0	0	1	0	1	0	1
0	26	0	1	2	0	0	0	1
1	26	1	0	2	1	1	0	1
1	26	0	1	2	0	0	1	0
1	22	0	1	1	1	0	1	1
0	28	0	0	2	0	1	0	1
1	29	0	1	2	0	0	0	1
0	30	0	0	2	0	0	0	1
1	30	1	0	2	0	0	1	1
0	30	0	0	2	1	0	0	1
1	22	0	0	2	0	0	1	1
1	25	1	0	2	0	1	0	1
1	27	1	0	2	0	0	0	1
0	27	0	1	2	0	1	0	1
1	25	1	1	2	1	0	1	1
0	34	0	0	2	0	1	0	1
0	20	1	0	2	0	0	0	1
0	30	0	0	2	0	0	0	1
0	25	1	1	2	0	0	1	1
1	30	0	0	2	1	1	0	1
0	28	0	0	2	0	0	0	1
1	27	1	1	2	1	0	1	1
1	28	0	0	2	0	0	0	1
0	24	0	1	1	0	0	1	1
1	24	1	0	2	1	0	1	1
0	35	0	0	2	0	1	0	1
0	25	1	1	2	0	0	1	1

1	25	0	1	1	0	0	0	1
0	27	0	0	2	1	1	0	1
1	26	1	0	2	0	1	1	1
1	24	0	1	2	1	0	0	1
0	24	0	0	2	0	0	1	1
0	24	0	1	2	1	0	0	1
1	24	0	1	2	0	1	0	1
1	26	1	0	2	0	0	1	1
0	25	1	0	2	1	1	1	1
1	27	0	1	2	0	0	0	1
0	20	1	0	1	0	1	0	0
1	21	0	1	2	0	0	0	1
0	20	0	1	1	0	0	1	1
1	22	1	1	2	0	0	0	1
0	26	0	0	2	0	0	1	1
0	23	0	0	2	1	1	0	1
1	24	0	1	2	0	1	0	1
1	22	0	0	2	1	0	1	0
0	23	0	1	2	1	0	1	1
0	22	1	1	2	0	0	0	1
0	24	0	0	2	1	0	0	1
1	22	0	1	2	0	1	1	1
0	25	0	1	2	1	0	0	1
1	19	0	0	1	0	1	0	1
0	24	1	1	2	1	0	0	1
0	26	0	1	1	0	1	0	1
0	31	0	0	1	1	0	1	1
0	30	0	0	2	0	0	0	1
0	27	0	0	2	1	0	1	1
1	27	1	0	1	0	1	0	1
1	27	0	1	1	0	0	0	1
1	28	1	0	2	0	1	1	1
0	31	0	1	1	1	1	0	0
1	26	1	0	1	0	0	0	1
1	32	0	0	1	0	1	0	1
0	23	0	0	2	0	0	0	0
1	27	0	1	1	1	0	0	1
1	26	0	0	2	0	0	0	1
1	27	0	0	1	1	0	0	1
1	26	0	0	1	0	1	1	0
1	28	0	0	2	0	0	0	1
0	27	0	0	2	1	0	0	1
1	31	1	0	2	0	0	0	1
1	27	0	0	2	1	0	0	1
0	26	1	0	2	1	1	0	1
1	27	0	1	2	0	1	0	1
0	26	0	0	2	0	0	0	1

1	25	0	0	2	1	0	0	1
1	25	1	0	2	0	1	1	1
1	20	0	0	1	1	0	1	0
0	26	0	1	2	0	0	0	0
0	23	1	0	2	0	0	1	0
1	25	1	0	2	0	1	0	0
0	30	0	0	1	0	0	0	0
1	20	0	0	1	0	0	1	1
0	24	1	0	1	0	0	0	1
1	25	0	0	2	0	1	0	1
1	21	0	0	1	1	0	1	1
0	27	0	0	1	1	0	0	1
1	28	1	1	2	0	1	0	0
1	25	0	0	1	0	0	0	1
0	27	0	1	1	1	0	0	0
1	25	0	0	2	1	0	0	1
0	26	1	1	2	0	0	1	0
1	28	0	0	1	1	1	0	0
1	25	0	1	2	1	0	0	1
1	24	1	0	2	0	0	1	0
1	25	0	0	2	1	0	0	1
0	24	0	1	1	0	1	1	0
1	27	1	0	2	0	0	0	1
1	25	0	1	2	1	1	0	0
0	24	0	0	2	0	0	0	1
0	27	0	1	2	1	0	1	1
1	27	1	0	2	0	0	1	1
1	24	0	1	1	1	1	0	1
1	27	0	0	2	1	0	0	1
0	28	0	1	2	0	1	1	0
0	25	1	0	2	1	0	0	1
1	26	0	1	1	0	0	0	1
0	28	0	0	2	0	1	1	1
1	27	0	1	2	0	0	1	1
0	27	1	0	2	1	0	0	1
1	25	0	0	2	0	1	1	1
1	25	1	1	2	0	0	0	0
1	27	0	1	1	0	1	1	1
1	25	1	0	2	1	1	0	1
1	28	0	1	2	0	0	0	1
1	19	0	0	1	0	1	1	1
0	34	0	0	2	0	0	0	0
1	20	1	0	1	1	1	1	1
0	20	0	1	1	0	0	0	0
0	28	1	0	1	1	1	1	0
0	32	1	0	2	0	0	0	1
1	22	0	0	1	1	1	1	1

1	24	0	0	1	0	0	1	0
1	20	1	1	1	1	1	1	1
1	21	0	1	1	0	0	0	1
0	32	0	0	1	0	0	1	0
1	26	1	1	2	0	1	1	1
1	27	0	1	1	1	1	0	1
1	35	0	1	2	0	0	1	0
0	27	0	0	2	0	1	0	1
0	30	0	0	2	0	0	0	0
1	31	1	1	2	0	1	1	0
0	24	0	0	2	1	0	0	1
1	35	0	1	2	0	1	1	1
1	31	1	0	2	1	0	1	1
1	25	0	0	2	0	1	0	1
1	37	1	0	2	1	0	0	1
0	23	0	0	2	0	0	1	1
0	27	0	1	2	1	1	0	1
0	26	0	0	1	0	0	0	1
1	27	0	0	2	1	1	0	1
1	24	1	0	2	0	0	1	1
1	28	0	1	1	1	1	0	0
0	27	1	1	2	0	0	1	0
1	34	0	0	2	0	0	1	1
1	27	0	0	2	1	1	0	0
0	20	1	0	1	1	0	0	0
1	18	0	1	1	0	1	1	0
1	35	1	0	2	0	0	0	1
1	25	1	1	2	0	1	1	0
1	27	0	0	1	1	1	1	1
0	27	0	1	1	0	0	0	0
1	24	1	0	2	1	1	1	1
1	28	0	1	2	0	0	0	1
0	29	1	0	2	1	1	1	1
1	20	0	0	1	0	1	0	1
0	20	0	1	1	0	0	0	0
1	25	1	0	2	1	0	0	1
1	18	1	0	1	0	1	0	1
1	22	1	1	1	1	0	1	1
0	24	0	1	2	0	1	0	1
0	31	0	0	2	0	0	0	1
1	23	1	1	1	1	0	1	0
1	22	0	0	1	0	0	0	0
1	42	1	0	2	1	0	0	1
1	22	0	1	1	0	1	0	0
0	27	0	0	1	1	0	0	0
0	32	0	1	1	0	0	0	0
0	40	1	0	2	1	1	0	1

1	24	0	0	1	0	1	1	1
1	23	0	1	1	1	1	1	1
1	26	0	1	2	0	0	0	0
1	27	1	1	2	0	0	0	0
1	24	0	0	1	0	1	0	0
0	28	0	0	2	1	1	1	1
1	27	0	1	2	0	1	1	0
1	23	0	0	1	0	0	0	0
0	28	0	0	2	0	1	0	1
0	32	0	1	2	1	0	1	0
1	27	1	0	2	0	1	1	0
1	25	0	0	2	0	0	0	0
0	30	0	0	2	0	1	0	1
0	28	0	1	1	0	0	0	1
1	24	0	0	2	1	1	0	0
1	28	1	1	2	1	0	1	0
1	24	0	0	2	0	0	0	1
1	19	1	1	1	0	0	0	0
1	19	0	0	1	0	1	0	1
1	33	1	0	2	1	0	0	1
1	22	0	0	2	1	1	1	0
1	22	1	0	2	1	0	0	1
0	25	0	1	1	0	1	1	0
1	17	0	0	1	1	1	1	1
1	43	0	0	1	0	0	0	1
0	42	1	1	1	1	1	1	0
1	52	0	0	1	0	1	0	0
1	32	1	1	1	0	1	1	0
1	25	0	0	1	0	0	0	0
1	24	0	1	1	1	1	1	0
0	19	1	1	1	0	0	0	0
0	48	0	1	1	0	0	0	0
1	25	1	0	2	1	1	1	0
0	26	0	1	1	0	0	0	0
0	27	1	0	2	1	0	0	1
1	23	0	1	2	0	1	1	0
1	23	0	1	2	1	1	1	0
0	25	1	1	2	1	0	0	1
1	17	0	0	2	0	0	1	0
1	36	1	0	1	0	0	1	0
0	25	0	1	1	1	0	0	0
1	32	1	0	1	0	1	1	0

Lampiran 4 Syntax Dan Output Program R-Studio

Visualisasi Data

```
library(readxl)
data3 <- read_excel("dataskripsivdfix.xlsx")
str(data3)
```

```
## tibble [234 × 17] (S3: tbl_df/tbl/data.frame)
##  $ X1 : num [1:234] 1 0 0 1 0 1 1 1 1 1 ...
##  $ X_1: chr [1:234] "Perempuan" "Laki-laki" "Laki-laki" "Perempuan" ...
##  $ X2 : num [1:234] 23 20 22 23 27 28 27 29 27 28 ...
##  $ X3 : num [1:234] 0 1 0 1 0 0 0 1 0 0 ...
##  $ X_3: chr [1:234] "Tidak Paham" "Paham" "Tidak Paham" "Paham" ...
##  $ X4 : num [1:234] 0 0 1 1 0 1 0 1 0 1 ...
##  $ X_4: chr [1:234] "Tidak Mendukung" "Tidak Mendukung" "Mendukung" "Mendukung" ..
##  $ X5 : num [1:234] 1 2 1 2 2 2 2 2 2 2 ...
##  $ X_5: chr [1:234] "SD/SMP/SMA" "S1/S2/S3" "SD/SMP/SMA" "S1/S2/S3" ...
##  $ X7 : num [1:234] 0 1 0 1 0 0 0 1 1 1 ...
##  $ X_7: chr [1:234] "Kurang" "Cukup" "Kurang" "Cukup" ...
##  $ X8 : num [1:234] 0 0 1 1 0 1 0 1 0 0 ...
##  $ X_8: chr [1:234] "Sedang" "Sedang" "Lama" "Lama" ...
##  $ X9 : num [1:234] 0 1 0 1 0 0 1 1 0 1 ...
##  $ X_9: chr [1:234] "Non Digital" "Digital" "Non Digital" "Digital" ...
##  $ Y  : num [1:234] 0 1 0 1 1 1 0 0 1 0 ...
##  $ Y_Y: chr [1:234] "Negatif" "Positif" "Negatif" "Positif" ...
```

```
y=as.factor(data3$Y)
x1=as.factor(data3$X1)
x2=(data3$X2)
x3=as.factor(data3$X3)
x4=as.factor(data3$X4)
x5=as.factor(data3$X5)
x7=as.factor(data3$X7)
x8=as.factor(data3$X8)
x9=as.factor(data3$X9)

Data3=data.frame(y,x1,x2,x3,x4,x5,x7,x8,x9)
str(Data3)
```

```
## 'data.frame':    234 obs. of  9 variables:
##  $ y : Factor w/ 5 levels "0","1","78","151",..: 1 2 1 2 2 2 1 1 2 1 ...
##  $ x1: Factor w/ 2 levels "0","1": 2 1 1 2 1 2 2 2 2 2 ...
##  $ x2: num  23 20 22 23 27 28 27 29 27 28 ...
##  $ x3: Factor w/ 4 levels "0","1","76","153": 1 2 1 2 1 1 1 2 1 1 ...
##  $ x4: Factor w/ 2 levels "0","1": 1 1 2 2 1 2 1 2 1 2 ...
##  $ x5: Factor w/ 2 levels "1","2": 1 2 1 2 2 2 2 2 2 2 ...
##  $ x7: Factor w/ 2 levels "0","1": 1 2 1 2 1 1 1 2 2 2 ...
##  $ x8: Factor w/ 2 levels "0","1": 1 1 2 2 1 2 1 2 1 1 ...
##  $ x9: Factor w/ 2 levels "0","1": 1 2 1 2 1 1 2 2 1 2 ...
```

```r
library(ggplot2)
library(scales)

# Sample Data Persepsi Masyarakat
jumlahPM1 = table(Data3$y)
kategori = c("Positif","Negatif")
nilai = c(151,78)
dataPM = data.frame(kategori, nilai)

# Persentase
persentase = (dataPM$nilai / sum(dataPM$nilai)) * 100

# Pie Chart Persepsi Masyarakat
PM = ggplot(dataPM,aes(x="",y = nilai, fill = kategori))+geom_bar(stat="identity", width =
1)+ coord_polar(theta = "y")+ geom_text(aes(label=paste0(round(persentase), "%")), positio
n = position_stack(vjust = 0.5))+
labs(x=NULL,y=NULL, title = "",fill="Kategori")+theme_classic()+theme(axis.line = element_
blank(),axis.text = element_blank(),axis.ticks = element_blank(),plot.title = element_text
(hjust=0.5,color="black"))
print(PM)

# Membuat Tabel Kontingensi
table(data3$X_1, data3$Y_Y)
```

```
##
##              Negatif Positif
##   Laki-laki       32      59
##   Perempuan       46      92
```

```r
tabel_kontingensi1 <- table(data3$Y_Y, data3$X_1)
ExpX1= chisq.test(tabel_kontingensi1)
ExpX1
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tabel_kontingensi1
## X-squared = 0.020654, df = 1, p-value = 0.8857
```

```r
ExpX1$expected
```

```
##
##          Laki-laki Perempuan
##   Negatif  30.99563  47.00437
##   Positif  60.00437  90.99563
```

```
# Mengubah Tabel Kontingensi menjadi Data Frame
df_kontingensi1 <- as.data.frame(tabel_kontingensi1)

# Menambahkan Nama Kolom
names(df_kontingensi1)= c("Persepsi","Jenis_Kelamin", "Jumlah")

# Membuat Barchart dengan ggplot2
barchart_ggplot = ggplot(df_kontingensi1, aes(x=Persepsi, y=Jumlah, fill=Jenis_Kelamin))+g
eom_bar(stat = "identity", position = "dodge")+ geom_text(aes(label =sprintf("%.2f%%", Jum
lah/sum(Jumlah)*100, "%")), position = position_dodge(width = 1), vjust = -0.5, color = "b
lack")+
labs(title = "",x="Persepsi", y="", fill="Jenis Kelamin")+
theme_minimal()+
theme(plot.title = element_text(hjust = 0.5, color = "black"))+scale_y_continuous(labels =
scales::percent_format(scale=0.4))

# Menampilkan Barchart
print(barchart_ggplot)
```

```
# Sample Data Usia
jumlahUsia = table(Data3$x2)
kategori2=c("17-27","28-38",">38")
nilai2 = c(165,57,7)
dataUsia=data.frame(kategori2, nilai2)

# Persentase
persentase2=(dataUsia$nilai2 / sum(dataUsia$nilai2))*100

# Pie Chart Persepsi Masyarakat
Usia= ggplot(dataUsia,aes(x="",y = nilai2, fill = kategori2))+geom_bar(stat="identity", wi
dth = 1)+ coord_polar(theta = "y")+ geom_text(aes(label=paste0(round(persentase2), "%")),
position = position_stack(vjust = 0.75))+
labs(x=NULL,y=NULL, title = "",fill="Kategori")+theme_classic()+theme(axis.line = element_
blank(),axis.text = element_blank(),axis.ticks = element_blank(),plot.title = element_text
(hjust=0.5,color="black"))+scale_fill_manual(values = c("aquamarine","mediumturquoise","co
ral"))
print(Usia)
```

```
# Membuat Tabel Kontingensi
table(data3$X_4, data3$Y_Y)
```

```
##
##                     Negatif Positif
##   Mendukung              42      50
##   Tidak Mendukung        36     101
```

```
tabel_kontingensi2 <- table(data3$Y_Y, data3$X_4)
ExpX2= chisq.test(tabel_kontingensi2)
ExpX2
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tabel_kontingensi2
## X-squared = 8.3567, df = 1, p-value = 0.003843
```

```
ExpX2$expected
```

```
##
##           Mendukung Tidak Mendukung
##   Negatif  31.33624        46.66376
##   Positif  60.66376        90.33624
```

```
# Mengubah Tabel Kontingensi menjadi Data Frame
df_kontingensi2 <- as.data.frame(tabel_kontingensi2)

# Menambahkan Nama Kolom
names(df_kontingensi2)= c("Persepsi","Perilaku", "Jumlah")

# Membuat Barchart dengan ggplot2
barchart_ggplot2 = ggplot(df_kontingensi2, aes(x=Persepsi, y=Jumlah, fill=Perilaku))+geom_
bar(stat = "identity", position = "dodge")+ geom_text(aes(label =sprintf("%.2f%%", Jumlah/
sum(Jumlah)*100, "%")), position = position_dodge(width = 1), vjust = -0.5, color = "blac
k")+
labs(title = "",x="Persepsi", y="", fill="Perilaku Masyarakat")+
theme_minimal()+
theme(plot.title = element_text(hjust = 0.5, color = "black"))+scale_y_continuous(labels =
scales::percent_format(scale=0.4))

# Menampilkan Barchart
print(barchart_ggplot2)
```

```
# Membuat Tabel Kontingensi
table(data3$X_5, data3$Y_Y)
```

```
##
##               Negatif Positif
##   S1/S2/S3        38     111
##   SD/SMP/SMA      40      40
```

```
tabel_kontingensi3 <- table(data3$Y_Y, data3$X_5)
ExpX3= chisq.test(tabel_kontingensi3)
ExpX3
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tabel_kontingensi3
## X-squared = 12.838, df = 1, p-value = 0.0003396
```

```
ExpX3$expected
```

```
##
##          S1/S2/S3 SD/SMP/SMA
##   Negatif 50.75109   27.24891
##   Positif 98.24891   52.75109
```

```
# Mengubah Tabel Kontingensi menjadi Data Frame
df_kontingensi3 <- as.data.frame(tabel_kontingensi3)

# Menambahkan Nama Kolom
names(df_kontingensi3)= c("Persepsi","Pendidikan", "Jumlah")

# Membuat Barchart dengan ggplot2
barchart_ggplot3 = ggplot(df_kontingensi3, aes(x=Persepsi, y=Jumlah, fill=Pendidikan))+geo
m_bar(stat = "identity", position = "dodge")+ geom_text(aes(label =sprintf("%.2f%%", Jumla
h/sum(Jumlah)*100, "%")), position = position_dodge(width = 1), vjust = -0.5, color = "bla
ck")+
labs(title = "",x="Persepsi", y="", fill="Pendidikan")+
theme_minimal()+
theme(plot.title = element_text(hjust = 0.5, color = "black"))+scale_y_continuous(labels =
scales::percent_format(scale=0.5))

# Menampilkan Barchart
print(barchart_ggplot3)
```

```
# Membuat Tabel Kontingensi
table(data3$X_7, data3$Y_Y)
```

```
##
##          Negatif Positif
##   Cukup       25      63
##   Kurang      53      88
```

```
tabel_kontingensi4 <- table(data3$Y_Y, data3$X_7)
ExpX4= chisq.test(tabel_kontingensi4)
ExpX4
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tabel_kontingensi4
## X-squared = 1.6447, df = 1, p-value = 0.1997
```

```
ExpX4$expected
```

```
##
##             Cukup  Kurang
##   Negatif 29.9738 48.0262
##   Positif 58.0262 92.9738
```

```
# Mengubah Tabel Kontingensi menjadi Data Frame
df_kontingensi4 <- as.data.frame(tabel_kontingensi4)

# Menambahkan Nama Kolom
names(df_kontingensi4)= c("Persepsi","Ekonomi", "Jumlah")

# Membuat Barchart dengan ggplot2
barchart_ggplot4 = ggplot(df_kontingensi4, aes(x=Persepsi, y=Jumlah, fill=Ekonomi))+geom_
ar(stat = "identity", position = "dodge")+ geom_text(aes(label =sprintf("%.2f%%", Jumlah/
um(Jumlah)*100, "%")), position = position_dodge(width = 1), vjust = -0.5, color = "blac
k")+
labs(title = "",x="Persepsi", y="", fill="Faktor Ekonomi")+
theme_minimal()+
theme(plot.title = element_text(hjust = 0.5, color = "black"))+scale_y_continuous(labels
scales::percent_format(scale=0.4))

# Menampilkan Barchart
print(barchart_ggplot4)
```

```r
# Membuat Tabel Kontingensi
table(data3$X_8, data3$Y_Y)
```

```
##
##          Negatif Positif
##   Lama        34      59
##   Sedang      44      92
```

```r
tabel_kontingensi5 <- table(data3$Y_Y, data3$X_8)
ExpX5= chisq.test(tabel_kontingensi5)
ExpX5
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tabel_kontingensi5
## X-squared = 0.26795, df = 1, p-value = 0.6047
```

```r
ExpX5$expected
```

```
##
##                Lama   Sedang
##   Negatif 31.67686 46.32314
##   Positif 61.32314 89.67686
```

```r
# Mengubah Tabel Kontingensi menjadi Data Frame
df_kontingensi5 <- as.data.frame(tabel_kontingensi5)

# Menambahkan Nama Kolom
names(df_kontingensi5)= c("Persepsi","Gadget", "Jumlah")

# Membuat Barchart dengan ggplot2
barchart_ggplot5 = ggplot(df_kontingensi5, aes(x=Persepsi, y=Jumlah, fill=Gadget))+geom_ba
r(stat = "identity", position = "dodge")+ geom_text(aes(label =sprintf("%.2f%%", Jumlah/su
m(Jumlah)*100, "%")), position = position_dodge(width = 1), vjust = -0.5, color = "black")
+
labs(title = "",x="Persepsi", y="", fill="Intensitas Bermain Gadget")+
theme_minimal()+
theme(plot.title = element_text(hjust = 0.5, color = "black"))+scale_y_continuous(labels =
scales::percent_format(scale=0.4))

# Menampilkan Barchart
print(barchart_ggplot5)
```

```
# Membuat Tabel Kontingensi
table(data3$X_9, data3$Y_Y)
```

```
##
##               Negatif Positif
##   Digital          37      52
##   Non Digital      41      99
```

```
tabel_kontingensi6 <- table(data3$Y_Y, data3$X_9)
ExpX6= chisq.test(tabel_kontingensi6)
ExpX6
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tabel_kontingensi6
## X-squared = 3.131, df = 1, p-value = 0.07682
```

```
ExpX6$expected
```

```
##
##            Digital Non Digital
##   Negatif 30.31441    47.68559
##   Positif 58.68559    92.31441
```

```
# Mengubah Tabel Kontingensi menjadi Data Frame
df_kontingensi6 <- as.data.frame(tabel_kontingensi6)

# Menambahkan Nama Kolom
names(df_kontingensi6)= c("Persepsi","Gaya", "Jumlah")

# Membuat Barchart dengan ggplot2
barchart_ggplot6 = ggplot(df_kontingensi6, aes(x=Persepsi, y=Jumlah, fill=Gaya))+geom_bar
(stat = "identity", position = "dodge")+ geom_text(aes(label =sprintf("%.2f%%", Jumlah/sum
(Jumlah)*100, "%")), position = position_dodge(width = 1), vjust = -0.5, color = "black")+
labs(title = "",x="Persepsi", y="", fill="Gaya Hidup")+
theme_minimal()+
theme(plot.title = element_text(hjust = 0.5, color = "black"))+scale_y_continuous(labels =
scales::percent_format(scale=0.4))

# Menampilkan Barchart
print(barchart_ggplot6)
```

```
# Membuat Tabel Kontingensi
table(data3$X_3, data3$Y_Y)
```

```
##
##                Negatif Positif
##   Paham             25      51
##   Tidak Paham       53     100
```

```
tabel_kontingensi7 <- table(data3$Y_Y, data3$X_3)
ExpX7= chisq.test(tabel_kontingensi7)
ExpX7
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tabel_kontingensi7
## X-squared = 0.013096, df = 1, p-value = 0.9089
```

```
ExpX7$expected
```

```
##
##              Paham Tidak Paham
##   Negatif 25.88646    52.11354
##   Positif 50.11354   100.88646
```

```
# Mengubah Tabel Kontingensi menjadi Data Frame
df_kontingensi7 <- as.data.frame(tabel_kontingensi7)

# Menambahkan Nama Kolom
names(df_kontingensi7)= c("Persepsi","Pemahaman", "Jumlah")

# Membuat Barchart dengan ggplot2
barchart_ggplot7 = ggplot(df_kontingensi7, aes(x=Persepsi, y=Jumlah, fill=Pemahaman))+geo
_bar(stat = "identity", position = "dodge")+ geom_text(aes(label =sprintf("%.2f%%", Jumla
h/sum(Jumlah)*100, "%")), position = position_dodge(width = 1), vjust = -0.5, color = "bl
ck")+
labs(title = "",x="Persepsi", y="", fill="Pemahaman Masyarakat")+
theme_minimal()+
theme(plot.title = element_text(hjust = 0.5, color = "black"))+scale_y_continuous(labels
scales::percent_format(scale=0.4))

# Menampilkan Barchart
print(barchart_ggplot7)
```

Visualisasi Data

```
library(readxl)
data1 <- read_excel("D:/Kuliah/Semester 7/Skripsi/data respon2.xlsx",
                    col_types = c("numeric","numeric","numeric","numeric","numeric","numeric","numeric",
                                  "numeric","numeric","numeric","numeric","numeric","numeric","numeric",
                                  "numeric","numeric","numeric","numeric","numeric","numeric","numeric",
                                  "numeric","numeric","numeric","numeric","numeric","numeric","numeric",
                                  "numeric","numeric","numeric","numeric","numeric","numeric","numeric",
                                  "numeric","numeric","numeric","numeric","numeric","numeric","numeric",
                                  "numeric","numeric","numeric","numeric","numeric","numeric","numeric",
                                  "numeric","numeric","numeric"))
```

Uji Validitas

#validitas Persepsi Masyarakat (Y)

```
valB1 = cor.test(data1$B1,data1$B)
valB1
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$B1 and data1$B
## t = 4.4721, df = 227, p-value = 1.224e-05
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.1608427 0.3994559
## sample estimates:
##       cor
## 0.2845505
```

```
valB2 = cor.test(data1$B2,data1$B)
valB2
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$B2 and data1$B
## t = 9.3139, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.4251682 0.6136365
## sample estimates:
##       cor
## 0.5258262
```

```
valB3 = cor.test(data1$B3,data1$B)
valB3
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$B3 and data1$B
## t = 6.8661, df = 227, p-value = 6.245e-11
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.3012406 0.5165577
## sample estimates:
##       cor
## 0.4146869
```

```
valB4 = cor.test(data1$B4,data1$B)
valB4
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$84 and data1$8
## t = 2.542, df = 227, p-value = 0.01169
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.03753691 0.28976003
## sample estimates:
##       cor
## 0.1663685
```

```
val85 = cor.test(data1$85,data1$8)
val85
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$85 and data1$8
## t = 7.1783, df = 227, p-value = 9.958e-12
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.3182212 0.5301950
## sample estimates:
##       cor
## 0.4301181
```

```
val86 = cor.test(data1$86,data1$8)
val86
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$86 and data1$8
## t = 5.2422, df = 227, p-value = 3.626e-07
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.2078252 0.4395291
## sample estimates:
##       cor
## 0.3286126
```

```
val87 = cor.test(data1$87,data1$8)
val87
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$87 and data1$8
## t = 8.7062, df = 227, p-value = 6.56e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.3963933 0.5915924
## sample estimates:
##       cor
## 0.5003234
```

```
val88 = cor.test(data1$88,data1$8)
val88
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$88 and data1$8
## t = 5.3316, df = 227, p-value = 2.348e-07
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.2131747 0.4440342
## sample estimates:
##       cor
## 0.3335066
```

```
val89 = cor.test(data1$89,data1$8)
val89
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$89 and data1$8
## t = 6.2918, df = 227, p-value = 1.598e-09
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.2691533 0.4904863
## sample estimates:
##       cor
## 0.3853483
```

```
val810 = cor.test(data1$810,data1$8)
val810
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$B10 and data1$B
## t = 10.491, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.4771196 0.6527055
## sample estimates:
##       cor
## 0.5714162
```

#Pemahaman dan Pengetahuan Masyarakat

```
valC1 = cor.test(data1$C1,data1$C)
valC1
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$C1 and data1$C
## t = 12.947, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.5702996 0.7205156
## sample estimates:
##       cor
## 0.6517538
```

```
valC2 = cor.test(data1$C2,data1$C)
valC2
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$C2 and data1$C
## t = 7.8246, df = 227, p-value = 1.917e-13
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.3522981 0.5572356
## sample estimates:
##       cor
## 0.4608894
```

```
valC3 = cor.test(data1$C3,data1$C)
valC3
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$C3 and data1$C
## t = 12.603, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.5584215 0.7120284
## sample estimates:
##       cor
## 0.6416135
```

```
valC4 = cor.test(data1$C4,data1$C)
valC4
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$C4 and data1$C
## t = 10.735, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.4872771 0.6602367
## sample estimates:
##       cor
## 0.5802624
```

```
valC5 = cor.test(data1$C5,data1$C)
valC5
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$C5 and data1$C
## t = 13.188, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.5783863 0.7262682
## sample estimates:
##       cor
## 0.6586408
```

#Perilaku Masyarakat

```
valD1 = cor.test(data1$D1,data1$D)
valD1
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$D1 and data1$D
## t = 3.9205, df = 227, p-value = 0.000117
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.1263115 0.3694099
## sample estimates:
##      cor
## 0.251829
```

```
valD2 = cor.test(data1$D2,data1$D)
valD2
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$D2 and data1$D
## t = 17.156, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.6888341 0.8028158
## sample estimates:
##       cor
## 0.7513764
```

```
valD3 = cor.test(data1$D3,data1$D)
valD3
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$D3 and data1$D
## t = 3.316, df = 227, p-value = 0.001063
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.08775117 0.33524626
## sample estimates:
##       cor
## 0.2149471
```

```
valD4 = cor.test(data1$D4,data1$D)
valD4
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$D4 and data1$D
## t = 18.391, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.7156848 0.8208742
## sample estimates:
##       cor
## 0.7735539
```

```
valD5 = cor.test(data1$D5,data1$D)
valD5
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$D5 and data1$D
## t = 2.5368, df = 227, p-value = 0.01186
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.03719272 0.28944427
## sample estimates:
##       cor
## 0.1660333
```

```
valD6 = cor.test(data1$D6,data1$D)
valD6
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$D6 and data1$D
## t = 18.876, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.7254282 0.8273702
## sample estimates:
##       cor
## 0.7815601
```

#Religiusitas

```
valE1 = cor.test(data1$E1,data1$E)
valE1
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$E1 and data1$E
## t = 11.444, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.5156649 0.6811015
## sample estimates:
##       cor
## 0.6048606
```

```
valE2 = cor.test(data1$E2,data1$E)
valE2
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$E2 and data1$E
## t = 12.6, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.5582994 0.7119409
## sample estimates:
##      cor
## 0.641509
```

```
valE3 = cor.test(data1$E3,data1$E)
valE3
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$E3 and data1$E
## t = 9.3556, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.4270936 0.6151012
## sample estimates:
##       cor
## 0.5275263
```

```
valE4 = cor.test(data1$E4,data1$E)
valE4
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$E4 and data1$E
## t = 7.7412, df = 227, p-value = 3.226e-13
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.3479824 0.5538349
## sample estimates:
##       cor
## 0.4570067
```

```
valE5 = cor.test(data1$E5,data1$E)
valE5
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$E5 and data1$E
## t = 9.3369, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.4262273 0.6144423
## sample estimates:
##       cor
## 0.5267614
```

```
valE6 = cor.test(data1$E6,data1$E)
valE6
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$E6 and data1$E
## t = 11.691, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.5251488 0.6880128
## sample estimates:
##       cor
## 0.6130528
```

#Faktor Ekonomi

```
valF1 = cor.test(data1$F1,data1$F)
valF1
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$F1 and data1$F
## t = 13.053, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.5738887 0.7230656
## sample estimates:
##       cor
## 0.6548053
```

```
valF2 = cor.test(data1$F2,data1$F)
valF2
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$F2 and data1$F
## t = 11.348, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.5119220 0.6783659
## sample estimates:
##      cor
## 0.601635
```

```
valF3 = cor.test(data1$F3,data1$F)
valF3
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$F3 and data1$F
## t = 12.271, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.5466096 0.7035438
## sample estimates:
##       cor
## 0.6315807
```

```
valF4 = cor.test(data1$F4,data1$F)
valF4
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$F4 and data1$F
## t = 14.465, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.6184433 0.7544687
## sample estimates:
##       cor
## 0.6925581
```

```
valF5 = cor.test(data1$F5,data1$F)
valF5
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$F5 and data1$F
## t = 10.896, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.4938881 0.6651197
## sample estimates:
##       cor
## 0.5860082
```

#Gaya Hidup

```
valG1 = cor.test(data1$G1,data1$G)
valG1
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$G1 and data1$G
## t = 15.637, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.6512709 0.7771970
## sample estimates:
##       cor
## 0.7201119
```

```
valG2 = cor.test(data1$G2,data1$G)
valG2
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$G2 and data1$G
## t = 16.373, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.6701388 0.7901177
## sample estimates:
##      cor
## 0.735851
```

```
valG3 = cor.test(data1$G3,data1$G)
valG3
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$G3 and data1$G
## t = 11.28, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.5092630 0.6764195
## sample estimates:
##       cor
## 0.5993352
```

```
valG4 = cor.test(data1$G4,data1$G)
valG4
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$G4 and data1$G
## t = 13.237, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.5799954 0.7274103
## sample estimates:
##       cor
## 0.6600096
```

```
valG5 = cor.test(data1$G5,data1$G)
valG5
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$G5 and data1$G
## t = 16.341, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.6693593 0.7895859
## sample estimates:
##       cor
## 0.7352822
```

#Intensitas Bermain Gadget

```
valH1 = cor.test(data1$H1,data1$H)
valH1
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$H1 and data1$H
## t = 13.358, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.5839890 0.7302416
## sample estimates:
##       cor
## 0.6634045
```

```
valH2 = cor.test(data1$H2,data1$H)
valH2
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$H2 and data1$H
## t = 8.0026, df = 227, p-value = 6.240e-14
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.3614242 0.5644042
## sample estimates:
##       cor
## 0.4690861
```

```
valH3 = cor.test(data1$H3,data1$H)
valH3
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$H3 and data1$H
## t = 8.6831, df = 227, p-value = 7.64e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.3952741 0.5907290
## sample estimates:
##       cor
## 0.4993278
```

```
valH4 = cor.test(data1$H4,data1$H)
valH4
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$H4 and data1$H
## t = 16.286, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.667979 0.788644
## sample estimates:
##      cor
## 0.734053
```

```
valH5 = cor.test(data1$H5,data1$H)
valH5
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$H5 and data1$H
## t = 18.133, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.7103402 0.8172964
## sample estimates:
##       cor
## 0.7691507
```

```
valH6 = cor.test(data1$H6,data1$H)
valH6
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$H6 and data1$H
## t = 5.4734, df = 227, p-value = 1.165e-07
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.2216208 0.4511233
## sample estimates:
##       cor
## 0.3414516
```

```
valH7 = cor.test(data1$H7,data1$H)
valH7
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$H7 and data1$H
## t = 18.292, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.7136582 0.8195132
## sample estimates:
##       cor
## 0.7718784
```

```
valH8 = cor.test(data1$H8,data1$H)
valH8
```

```
##
##  Pearson's product-moment correlation
##
## data:  data1$H8 and data1$H
## t = 17.465, df = 227, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.6958472 0.8075527
## sample estimates:
##       cor
## 0.7571826
```

Uji Reliabilitas

```
library(psych)
attach(data1)
```

```
## The following object is masked from package:base:
##
##     F
```

#Persepsi Masyarakat

```
b = data.frame(B1,B2,B3,B4,B5,B6,B7,B8,B9,B10)
rel1 = alpha(b)
```

```
## Warning in alpha(b): Some items were negatively correlated with the total scale and probably
## should be reversed.
## To do this, run the function again with the 'check.keys=TRUE' option
```

```
## Some items ( B8 B9 ) were negatively correlated with the total scale and
## probably should be reversed.
## To do this, run the function again with the 'check.keys=TRUE' option
```

```
rel1
```

```
##
## Reliability analysis
## Call: alpha(x = b)
##
##   raw_alpha std.alpha G6(smc) average_r  S/N   ase mean   sd median_r
##       0.34      0.43    0.51     0.069 0.74 0.066  3.1 0.27    0.046
##
##     95% confidence boundaries
##          lower alpha upper
## Feldt     0.21  0.34  0.46
## Duhachek  0.21  0.34  0.47
##
##  Reliability if an item is dropped:
##     raw_alpha std.alpha G6(smc) average_r  S/N alpha se var.r med.r
## B1       0.35      0.44    0.52     0.080 0.78    0.066 0.034 0.049
## B2       0.24      0.32    0.42     0.049 0.47    0.077 0.029 0.033
## B3       0.29      0.37    0.46     0.060 0.58    0.072 0.031 0.025
## B4       0.42      0.48    0.55     0.094 0.93    0.059 0.030 0.063
## B5       0.28      0.36    0.46     0.060 0.57    0.072 0.031 0.019
## B6       0.32      0.40    0.50     0.070 0.68    0.069 0.034 0.038
## B7       0.25      0.33    0.42     0.051 0.49    0.075 0.027 0.024
## B8       0.38      0.48    0.52     0.093 0.92    0.063 0.025 0.087
## B9       0.41      0.47    0.52     0.091 0.90    0.058 0.026 0.087
## B10      0.21      0.29    0.40     0.044 0.41    0.079 0.028 0.019
##
##  Item statistics
##        n raw.r std.r  r.cor  r.drop mean   sd
## B1   229  0.28  0.31  0.116  0.0454  3.4 0.66
## B2   229  0.53  0.58  0.554  0.3413  3.5 0.60
## B3   229  0.41  0.48  0.395  0.2143  3.4 0.59
## B4   229  0.17  0.18 -0.045 -0.1054  3.2 0.74
## B5   229  0.43  0.49  0.402  0.2141  3.4 0.64
## B6   229  0.33  0.39  0.232  0.1201  3.5 0.59
## B7   229  0.50  0.56  0.536  0.2804  3.3 0.68
## B8   229  0.33  0.19  0.024  0.0123  2.0 0.88
## B9   229  0.39  0.21  0.047 -0.0027  2.2 1.06
## B10  229  0.57  0.63  0.626  0.3830  3.4 0.63
##
## Non missing response frequency for each item
```

```
##         1    2    3    4 miss
## B1   0.02 0.03 0.49 0.45    0
## B2   0.00 0.05 0.44 0.51    0
## B3   0.01 0.03 0.51 0.45    0
## B4   0.04 0.07 0.52 0.36    0
## B5   0.00 0.07 0.48 0.44    0
## B6   0.01 0.02 0.47 0.50    0
## B7   0.01 0.08 0.47 0.44    0
## B8   0.31 0.47 0.14 0.08    0
## B9   0.31 0.33 0.19 0.17    0
## B10  0.01 0.06 0.48 0.45    0
```

```
b2 = data.frame(B1,B2,B3,B4,B5,B6,B7,B8,B10)
rel12 = alpha(b2)
```

```
## Warning in alpha(b2): Some items were negatively correlated with the total scale and probably
## should be reversed.
## To do this, run the function again with the 'check.keys=TRUE' option
```

```
## Some items ( B8 ) were negatively correlated with the total scale and
## probably should be reversed.
## To do this, run the function again with the 'check.keys=TRUE' option
```

```
rel12
```

```
##
## Reliability analysis
## Call: alpha(x = b2)
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean   sd median_r
##       0.41      0.47    0.52     0.091 0.9 0.058  3.2 0.28    0.087
##
##     95% confidence boundaries
##          lower alpha upper
## Feldt     0.29  0.41  0.52
## Duhachek  0.29  0.41  0.52
##
##  Reliability if an item is dropped:
##     raw_alpha std.alpha G6(smc) average_r  S/N alpha se var.r med.r
## B1       0.42      0.49    0.53     0.108 0.96    0.058 0.029 0.123
## B2       0.30      0.37    0.42     0.070 0.60    0.070 0.023 0.074
## B3       0.34      0.41    0.46     0.080 0.70    0.066 0.026 0.062
## B4       0.46      0.52    0.54     0.119 1.08    0.054 0.025 0.146
## B5       0.34      0.41    0.46     0.081 0.70    0.066 0.026 0.061
## B6       0.38      0.45    0.50     0.094 0.83    0.062 0.030 0.087
## B7       0.27      0.36    0.41     0.065 0.55    0.074 0.024 0.062
## B8       0.56      0.57    0.58     0.142 1.33    0.044 0.019 0.147
## B10      0.26      0.34    0.39     0.060 0.51    0.074 0.023 0.062
##
##  Item statistics
##        n raw.r std.r  r.cor r.drop mean   sd
## B1   229  0.32 0.320  0.124  0.059  3.4 0.66
## B2   229  0.57 0.589  0.558  0.373  3.5 0.60
## B3   229  0.48 0.512  0.424  0.271  3.4 0.59
## B4   229  0.27 0.240  0.021 -0.028  3.2 0.74
## B5   229  0.48 0.509  0.420  0.257  3.4 0.64
## B6   229  0.38 0.419  0.258  0.160  3.5 0.59
## B7   229  0.63 0.625  0.601  0.416  3.3 0.68
## B8   229  0.16 0.071 -0.218 -0.189  2.0 0.88
## B10  229  0.64 0.657  0.653  0.452  3.4 0.63
##
## Non missing response frequency for each item
##        1    2    3    4 miss
## B1  0.02 0.03 0.49 0.45    0
```

```
## B2  0.00 0.05 0.44 0.51    0
## B3  0.01 0.03 0.51 0.45    0
## B4  0.04 0.07 0.52 0.36    0
## B5  0.00 0.07 0.48 0.44    0
## B6  0.01 0.02 0.47 0.50    0
## B7  0.01 0.08 0.47 0.44    0
## B8  0.31 0.47 0.14 0.08    0
## B10 0.01 0.06 0.48 0.45    0
```

```
b3 = data.frame(B1,B2,B3,B4,B5,B6,B7,B10)
rel13 = alpha(b3)
rel13
```

```
##
## Reliability analysis
## Call: alpha(x = b3)
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean   sd median_r
##       0.56      0.57    0.58      0.14 1.3 0.044  3.4 0.32     0.15
##
##    95% confidence boundaries
##          lower alpha upper
## Feldt     0.47  0.56  0.64
## Duhachek  0.47  0.56  0.65
##
##  Reliability if an item is dropped:
##     raw_alpha std.alpha G6(smc) average_r  S/N alpha se var.r med.r
## B1       0.58      0.59    0.59      0.17 1.44    0.042 0.020  0.19
## B2       0.49      0.50    0.51      0.13 1.02    0.051 0.015  0.13
## B3       0.53      0.54    0.54      0.14 1.16    0.047 0.018  0.13
## B4       0.61      0.61    0.60      0.18 1.56    0.039 0.016  0.19
## B5       0.50      0.52    0.53      0.13 1.07    0.050 0.022  0.13
## B6       0.55      0.56    0.57      0.16 1.29    0.045 0.022  0.16
## B7       0.47      0.48    0.50      0.12 0.94    0.054 0.018  0.13
## B10      0.46      0.47    0.48      0.11 0.88    0.055 0.018  0.12
##
##  Item statistics
##       n raw.r std.r r.cor r.drop mean   sd
## B1  229  0.36  0.35 0.158  0.111  3.4 0.66
## B2  229  0.56  0.58 0.519  0.371  3.5 0.60
## B3  229  0.47  0.50 0.392  0.266  3.4 0.59
## B4  229  0.34  0.29 0.078  0.049  3.2 0.74
## B5  229  0.55  0.55 0.447  0.341  3.4 0.64
## B6  229  0.41  0.43 0.262  0.189  3.5 0.59
## B7  229  0.64  0.63 0.572  0.431  3.3 0.68
## B10 229  0.66  0.66 0.626  0.475  3.4 0.63
##
## Non missing response frequency for each item
##        1    2    3    4 miss
## B1  0.02 0.03 0.49 0.45    0
## B2  0.00 0.05 0.44 0.51    0
## B3  0.01 0.03 0.51 0.45    0
## B4  0.04 0.07 0.52 0.36    0
## B5  0.00 0.07 0.48 0.44    0
## B6  0.01 0.02 0.47 0.50    0
## B7  0.01 0.08 0.47 0.44    0
## B10 0.01 0.06 0.48 0.45    0
```

```
b4 = data.frame(B1,B2,B3,B5,B6,B7,B10)
rel14 = alpha(b4)
rel14
```

```
##
## Reliability analysis
## Call: alpha(x = b4)
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean   sd median_r
##       0.61      0.61     0.6      0.18 1.6 0.039  3.4 0.34     0.19
##
##     95% confidence boundaries
##          lower alpha upper
## Feldt     0.53  0.61  0.68
## Duhachek  0.53  0.61  0.68
##
##  Reliability if an item is dropped:
##     raw_alpha std.alpha G6(smc) average_r S/N alpha se  var.r med.r
## B1       0.65      0.65    0.62      0.24 1.8    0.035 0.0085  0.22
## B2       0.55      0.55    0.53      0.17 1.2    0.046 0.0109  0.17
## B3       0.56      0.56    0.56      0.18 1.3    0.045 0.0198  0.19
## B5       0.56      0.57    0.56      0.18 1.3    0.044 0.0190  0.19
## B6       0.60      0.61    0.60      0.20 1.5    0.040 0.0183  0.22
## B7       0.54      0.54    0.53      0.16 1.2    0.047 0.0140  0.19
## B10      0.51      0.51    0.50      0.15 1.0    0.050 0.0157  0.17
##
##  Item statistics
##        n raw.r std.r r.cor r.drop mean   sd
## B1  229  0.36  0.34  0.13  0.085  3.4 0.66
## B2  229  0.59  0.60  0.52  0.389  3.5 0.60
## B3  229  0.56  0.57  0.45  0.353  3.4 0.59
## B5  229  0.56  0.56  0.43  0.340  3.4 0.64
## B6  229  0.44  0.46  0.28  0.217  3.5 0.59
## B7  229  0.63  0.62  0.54  0.413  3.3 0.68
## B10 229  0.68  0.68  0.63  0.496  3.4 0.63
##
## Non missing response frequency for each item
##        1    2    3    4 miss
## B1  0.02 0.03 0.49 0.45    0
## B2  0.00 0.05 0.44 0.51    0
## B3  0.01 0.03 0.51 0.45    0
## B5  0.00 0.07 0.48 0.44    0
## B6  0.01 0.02 0.47 0.50    0
```

```
## B7  0.01 0.08 0.47 0.44    0
## B10 0.01 0.06 0.48 0.45    0
```

```
c = data.frame(C1,C2,C3,C4,C5)
rel2 = alpha(c)
rel2
```

```
##
## Reliability analysis
## Call: alpha(x = c)
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean   sd median_r
##       0.55      0.55    0.55       0.2 1.2 0.047  3.4 0.34     0.22
##
##     95% confidence boundaries
##          lower alpha upper
## Feldt     0.46  0.55  0.64
## Duhachek  0.46  0.55  0.65
##
##  Reliability if an item is dropped:
##    raw_alpha std.alpha G6(smc) average_r  S/N alpha se var.r med.r
## C1      0.45      0.45    0.43      0.17 0.83    0.059 0.016  0.17
## C2      0.58      0.58    0.53      0.26 1.40    0.045 0.012  0.27
## C3      0.47      0.47    0.44      0.18 0.90    0.057 0.013  0.21
## C4      0.52      0.52    0.50      0.21 1.07    0.052 0.034  0.28
## C5      0.45      0.44    0.42      0.16 0.79    0.059 0.029  0.17
##
##  Item statistics
##      n raw.r std.r r.cor r.drop mean   sd
## C1 229  0.65  0.65  0.54   0.39  3.3 0.56
## C2 229  0.46  0.48  0.25   0.16  3.3 0.55
## C3 229  0.64  0.63  0.51   0.36  3.4 0.60
## C4 229  0.58  0.57  0.37   0.28  3.4 0.59
## C5 229  0.66  0.67  0.55   0.40  3.3 0.56
##
## Non missing response frequency for each item
##       1    2    3    4 miss
## C1 0.00 0.05 0.58 0.38    0
## C2 0.00 0.03 0.60 0.37    0
## C3 0.00 0.05 0.50 0.45    0
## C4 0.01 0.03 0.55 0.41    0
## C5 0.00 0.03 0.58 0.39    0
```

```
c3 = data.frame(C1,C3,C4,C5)
rel23 = alpha(c3)
rel23
```

```
##
## Reliability analysis
## Call: alpha(x = c3)
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean   sd median_r
##       0.58      0.58    0.53      0.26 1.4 0.045  3.4 0.39     0.27
##
##     95% confidence boundaries
##          lower alpha upper
## Feldt     0.49  0.58  0.66
## Duhachek  0.49  0.58  0.67
##
##  Reliability if an item is dropped:
##    raw_alpha std.alpha G6(smc) average_r  S/N alpha se  var.r med.r
## C1      0.43      0.43    0.35      0.20 0.76    0.065 0.0110  0.17
## C3      0.45      0.45    0.36      0.22 0.82    0.063 0.0076  0.26
## C4      0.60      0.60    0.51      0.33 1.50    0.046 0.0066  0.32
## C5      0.55      0.55    0.47      0.29 1.21    0.052 0.0155  0.27
##
##  Item statistics
##      n raw.r std.r r.cor r.drop mean   sd
## C1 229  0.73  0.73  0.61   0.47  3.3 0.56
## C3 229  0.72  0.72  0.59   0.44  3.4 0.60
## C4 229  0.59  0.58  0.33   0.25  3.4 0.59
## C5 229  0.63  0.63  0.42   0.32  3.3 0.56
##
## Non missing response frequency for each item
##       1    2    3    4 miss
## C1 0.00 0.05 0.58 0.38    0
## C3 0.00 0.05 0.50 0.45    0
## C4 0.01 0.03 0.55 0.41    0
## C5 0.00 0.03 0.58 0.39    0
```

```r
c4 = data.frame(C1,C3,C5)
rel24 = alpha(c4)
rel24
```

```
##
## Reliability analysis
## Call: alpha(x = c4)
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean   sd median_r
##        0.6       0.6    0.51      0.33 1.5 0.046  3.4 0.43     0.32
##
##     95% confidence boundaries
##          lower alpha upper
## Feldt     0.50   0.6  0.68
## Duhachek  0.51   0.6  0.69
##
##  Reliability if an item is dropped:
##    raw_alpha std.alpha G6(smc) average_r  S/N alpha se var.r med.r
## C1      0.48      0.48    0.32      0.32 0.93    0.068    NA  0.32
## C3      0.41      0.41    0.26      0.26 0.70    0.078    NA  0.26
## C5      0.59      0.59    0.42      0.42 1.45    0.054    NA  0.42
##
##  Item statistics
##       n raw.r std.r r.cor r.drop mean   sd
## C1 229  0.75  0.75  0.55   0.42  3.3 0.56
## C3 229  0.79  0.78  0.60   0.47  3.4 0.60
## C5 229  0.70  0.71  0.44   0.34  3.3 0.56
##
## Non missing response frequency for each item
##    1    2    3    4 miss
## C1 0 0.05 0.58 0.38    0
## C3 0 0.05 0.50 0.45    0
## C5 0 0.03 0.58 0.39    0
```

```r
d = data.frame(D1,D2,D3,D4,D5,D6)
rel3 = alpha(d)
```

```
## Warning in alpha(d): Some items were negatively correlated with the total scale and probably
## should be reversed.
## To do this, run the function again with the 'check.keys=TRUE' option
```

```
## Some items ( D1 D3 D5 ) were negatively correlated with the total scale and
## probably should be reversed.
## To do this, run the function again with the 'check.keys=TRUE' option
```

```r
rel3
```

```
##
## Reliability analysis
## Call: alpha(x = d)
##
##   raw_alpha std.alpha G6(smc) average_r  S/N  ase mean   sd median_r
##       0.54      0.45    0.54      0.12 0.81 0.04  2.9 0.46   -0.053
##
##    95% confidence boundaries
##          lower alpha upper
## Feldt     0.44  0.54  0.63
## Duhachek  0.46  0.54  0.62
##
##  Reliability if an item is dropped:
##    raw_alpha std.alpha G6(smc) average_r  S/N alpha se var.r  med.r
## D1      0.57      0.45    0.54     0.140 0.81    0.036 0.100 -0.060
## D2      0.36      0.33    0.43     0.089 0.49    0.060 0.061 -0.061
## D3      0.58      0.47    0.56     0.151 0.89    0.035 0.097 -0.034
## D4      0.34      0.32    0.40     0.087 0.48    0.063 0.058 -0.032
## D5      0.59      0.49    0.57     0.164 0.98    0.034 0.092 -0.032
## D6      0.33      0.31    0.38     0.082 0.45    0.064 0.058 -0.032
##
##  Item statistics
##       n raw.r std.r r.cor r.drop mean   sd
## D1 229  0.25  0.45  0.22  0.054  3.4 0.54
## D2 229  0.75  0.61  0.56  0.495  2.1 1.03
## D3 229  0.21  0.41  0.16  0.013  3.4 0.55
## D4 229  0.77  0.62  0.60  0.521  2.4 1.06
## D5 229  0.17  0.37  0.11 -0.030  3.4 0.53
## D6 229  0.78  0.63  0.63  0.547  2.5 1.02
##
## Non missing response frequency for each item
##       1    2    3    4 miss
## D1 0.00 0.02 0.58 0.40    0
## D2 0.34 0.41 0.09 0.16    0
## D3 0.01 0.00 0.54 0.45    0
## D4 0.23 0.29 0.28 0.20    0
## D5 0.00 0.01 0.57 0.42    0
## D6 0.21 0.27 0.34 0.19    0
```

```
d2 = data.frame(D1,D2,D3,D4,D6)
rel32 = alpha(d2)
```

```
## Warning in alpha(d2): Some items were negatively correlated with the total scale and probably
## should be reversed.
## To do this, run the function again with the 'check.keys=TRUE' option
```

```
## Some items ( D1 D3 ) were negatively correlated with the total scale and
## probably should be reversed.
## To do this, run the function again with the 'check.keys=TRUE' option
```

```
rel32
```

```
##
## Reliability analysis
## Call: alpha(x = d2)
##
##   raw_alpha std.alpha G6(smc) average_r  S/N   ase mean   sd median_r
##       0.59      0.49    0.57      0.16 0.98 0.034  2.8 0.54   -0.032
##
##     95% confidence boundaries
##          lower alpha upper
## Feldt     0.50  0.59  0.67
## Duhachek  0.53  0.59  0.66
##
##  Reliability if an item is dropped:
##    raw_alpha std.alpha G6(smc) average_r  S/N alpha se var.r  med.r
## D1      0.66      0.57    0.61     0.250 1.33    0.029 0.127  0.241
## D2      0.40      0.31    0.42     0.102 0.45    0.054 0.087 -0.061
## D3      0.67      0.59    0.62     0.265 1.44    0.029 0.116  0.263
## D4      0.39      0.32    0.39     0.105 0.47    0.057 0.065 -0.032
## D6      0.37      0.30    0.37     0.097 0.43    0.058 0.064 -0.032
##
##  Item statistics
##       n raw.r std.r r.cor  r.drop mean   sd
## D1 229  0.21  0.40 0.098  0.0044  3.4 0.54
## D2 229  0.79  0.70 0.640  0.5512  2.1 1.03
## D3 229  0.17  0.36 0.059 -0.0319  3.4 0.55
## D4 229  0.80  0.70 0.673  0.5684  2.4 1.06
## D6 229  0.81  0.71 0.701  0.5946  2.5 1.02
##
## Non missing response frequency for each item
##       1    2    3    4 miss
## D1 0.00 0.02 0.58 0.40    0
## D2 0.34 0.41 0.09 0.16    0
## D3 0.01 0.00 0.54 0.45    0
## D4 0.23 0.29 0.28 0.20    0
## D6 0.21 0.27 0.34 0.19    0
```

```
d3 = data.frame(D1,D2,D4,D6)
rel33 = alpha(d3)
```

```
## Warning in alpha(d3): Some items were negatively correlated with the total scale and probably
## should be reversed.
## To do this, run the function again with the 'check.keys=TRUE' option
```

```
## Some items ( D1 ) were negatively correlated with the total scale and
## probably should be reversed.
## To do this, run the function again with the 'check.keys=TRUE' option
```

```
rel33
```

```
##
## Reliability analysis
## Call: alpha(x = d3)
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean   sd median_r
##       0.67      0.59    0.62      0.26 1.4 0.029  2.6 0.66     0.26
##
##    95% confidence boundaries
##          lower alpha upper
## Feldt     0.59  0.67  0.73
## Duhachek  0.61  0.67  0.73
##
##  Reliability if an item is dropped:
##    raw_alpha std.alpha G6(smc) average_r  S/N alpha se  var.r   med.r
## D1      0.80      0.80    0.73      0.57 4.04    0.023 0.0027  0.5536
## D2      0.50      0.38    0.45      0.17 0.61    0.044 0.1608 -0.0585
## D4      0.47      0.36    0.40      0.16 0.57    0.049 0.1170 -0.0098
## D6      0.46      0.36    0.39      0.16 0.55    0.050 0.1088 -0.0098
##
##  Item statistics
##      n raw.r std.r r.cor r.drop mean   sd
## D1 229  0.15  0.32 -0.06 -0.052  3.4 0.54
## D2 229  0.81  0.78  0.68  0.587  2.1 1.03
## D4 229  0.84  0.79  0.74  0.625  2.4 1.06
## D6 229  0.84  0.79  0.75  0.638  2.5 1.02
##
## Non missing response frequency for each item
##       1    2    3    4 miss
## D1 0.00 0.02 0.58 0.40    0
## D2 0.34 0.41 0.09 0.16    0
## D4 0.23 0.29 0.28 0.20    0
## D6 0.21 0.27 0.34 0.19    0
```

```
e = data.frame(E1,E2,E3,E4,E5,E6)
rel4 = alpha(e)
rel4
```

```
##
## Reliability analysis
## Call: alpha(x = e)
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean   sd median_r
##       0.57      0.57    0.55      0.18 1.3 0.044  3.4 0.33     0.19
##
##     95% confidence boundaries
##          lower alpha upper
## Feldt     0.48  0.57  0.65
## Duhachek  0.48  0.57  0.66
##
##  Reliability if an item is dropped:
##    raw_alpha std.alpha G6(smc) average_r  S/N alpha se  var.r med.r
## E1      0.50      0.50    0.47      0.17 1.01    0.052 0.0107  0.16
## E2      0.49      0.48    0.45      0.16 0.94    0.054 0.0096  0.13
## E3      0.54      0.54    0.51      0.19 1.19    0.047 0.0078  0.18
## E4      0.57      0.57    0.52      0.21 1.32    0.045 0.0046  0.21
## E5      0.54      0.54    0.50      0.19 1.17    0.048 0.0094  0.21
## E6      0.49      0.48    0.45      0.16 0.94    0.053 0.0104  0.12
##
##  Item statistics
##      n raw.r std.r r.cor r.drop mean   sd
## E1 229  0.60  0.60  0.46   0.35  3.4 0.60
## E2 229  0.64  0.62  0.51   0.39  3.3 0.63
## E3 229  0.53  0.52  0.35   0.26  3.3 0.58
## E4 229  0.46  0.47  0.28   0.19  3.4 0.55
## E5 229  0.53  0.53  0.36   0.27  3.3 0.56
## E6 229  0.61  0.63  0.52   0.39  3.5 0.55
##
## Non missing response frequency for each item
##       1    2    3    4 miss
## E1 0.01 0.03 0.51 0.45    0
## E2 0.00 0.07 0.51 0.41    0
## E3 0.00 0.05 0.59 0.35    0
## E4 0.00 0.02 0.55 0.42    0
## E5 0.00 0.03 0.59 0.38    0
## E6 0.00 0.01 0.46 0.52    0
```

```
e2 = data.frame(E1,E2,E3,E5,E6)
re24 = alpha(e2)
re24
```

```
##
## Reliability analysis
## Call: alpha(x = e2)
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean   sd median_r
##       0.57      0.57    0.52      0.21 1.3 0.045  3.4 0.35     0.21
##
##    95% confidence boundaries
##          lower alpha upper
## Feldt     0.48  0.57  0.65
## Duhachek  0.48  0.57  0.66
##
##  Reliability if an item is dropped:
##    raw_alpha std.alpha G6(smc) average_r  S/N alpha se   var.r med.r
## E1      0.50      0.50    0.44      0.20 0.99    0.054 0.00531  0.20
## E2      0.47      0.47    0.41      0.18 0.89    0.057 0.00445  0.16
## E3      0.53      0.53    0.47      0.22 1.12    0.050 0.00548  0.23
## E5      0.54      0.55    0.48      0.23 1.20    0.049 0.00099  0.23
## E6      0.52      0.52    0.46      0.21 1.09    0.051 0.00678  0.23
##
##  Item statistics
##      n raw.r std.r r.cor r.drop mean   sd
## E1 229  0.63  0.63  0.47   0.36  3.4 0.60
## E2 229  0.67  0.66  0.53   0.40  3.3 0.63
## E3 229  0.59  0.59  0.40   0.30  3.3 0.58
## E5 229  0.55  0.56  0.37   0.28  3.3 0.56
## E6 229  0.58  0.59  0.41   0.31  3.5 0.55
##
## Non missing response frequency for each item
##       1    2    3    4 miss
## E1 0.01 0.03 0.51 0.45    0
## E2 0.00 0.07 0.51 0.41    0
## E3 0.00 0.05 0.59 0.35    0
## E5 0.00 0.03 0.59 0.38    0
## E6 0.00 0.01 0.46 0.52    0
```

```
f = data.frame(F1,F2,F3,F4,F5)
rel5 = alpha(f)
rel5
```

```
##
## Reliability analysis
## Call: alpha(x = f)
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean   sd median_r
##       0.62      0.63    0.59      0.25 1.7 0.039  3.3 0.43     0.23
##
##     95% confidence boundaries
##          lower alpha upper
## Feldt     0.54  0.62   0.7
## Duhachek  0.55  0.62   0.7
##
##  Reliability if an item is dropped:
##    raw_alpha std.alpha G6(smc) average_r S/N alpha se  var.r med.r
## F1      0.54      0.55    0.48      0.23 1.2    0.049 0.0029  0.23
## F2      0.57      0.58    0.53      0.26 1.4    0.046 0.0088  0.23
## F3      0.56      0.56    0.50      0.24 1.3    0.047 0.0035  0.22
## F4      0.55      0.56    0.50      0.24 1.3    0.049 0.0085  0.21
## F5      0.62      0.62    0.56      0.29 1.6    0.041 0.0059  0.29
##
##  Item statistics
##      n raw.r std.r r.cor r.drop mean   sd
## F1 229  0.65  0.67  0.56   0.44  3.3 0.62
## F2 229  0.60  0.62  0.46   0.37  3.3 0.62
## F3 229  0.63  0.65  0.52   0.39  3.3 0.65
## F4 229  0.69  0.66  0.53   0.42  3.2 0.79
## F5 229  0.59  0.56  0.36   0.29  3.3 0.74
##
## Non missing response frequency for each item
##       1    2    3    4 miss
## F1 0.00 0.07 0.51 0.42    0
## F2 0.01 0.06 0.55 0.38    0
## F3 0.03 0.04 0.59 0.35    0
## F4 0.03 0.13 0.43 0.41    0
## F5 0.03 0.08 0.44 0.45    0

g = data.frame(G1,G2,G3,G4,G5)
rel6 = alpha(g)
rel6
```

```
## 
## Reliability analysis
## Call: alpha(x = g)
## 
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean  sd median_r
##       0.73      0.73    0.73      0.35 2.6 0.029  2.5 0.6     0.25
## 
##     95% confidence boundaries
##          lower alpha upper
## Feldt     0.67  0.73  0.78
## Duhachek  0.67  0.73  0.78
## 
##  Reliability if an item is dropped:
##    raw_alpha std.alpha G6(smc) average_r S/N alpha se var.r med.r
## G1      0.66      0.66    0.67      0.33 2.0    0.037 0.036  0.24
## G2      0.66      0.66    0.65      0.33 1.9    0.038 0.029  0.25
## G3      0.72      0.72    0.68      0.39 2.5    0.030 0.025  0.37
## G4      0.70      0.69    0.66      0.36 2.2    0.032 0.036  0.35
## G5      0.66      0.66    0.64      0.33 1.9    0.038 0.024  0.25
## 
##  Item statistics
##      n raw.r std.r r.cor r.drop mean   sd
## G1 229  0.72  0.72  0.62   0.53  3.1 0.85
## G2 229  0.74  0.72  0.64   0.54  2.9 0.98
## G3 229  0.60  0.62  0.49   0.38  1.8 0.80
## G4 229  0.66  0.67  0.56   0.44  1.9 0.87
## G5 229  0.74  0.72  0.65   0.54  3.0 0.90
## 
## Non missing response frequency for each item
##       1    2    3    4 miss
## G1 0.07 0.13 0.48 0.33    0
## G2 0.10 0.14 0.48 0.28    0
## G3 0.39 0.47 0.10 0.05    0
## G4 0.39 0.45 0.09 0.07    0
## G5 0.10 0.09 0.52 0.29    0
```

```
h = data.frame(H1,H2,H3,H4,H5,H6,H7,H8)
rel7 = alpha(h)
rel7
```

```
##
## Reliability analysis
## Call: alpha(x = h)
##
##   raw_alpha std.alpha G6(smc) average_r S/N  ase mean   sd median_r
##        0.8      0.78     0.8      0.31 3.6 0.019    3 0.56     0.24
##
##    95% confidence boundaries
##          lower alpha upper
## Feldt     0.75   0.8  0.83
## Duhachek  0.76   0.8  0.83
##
##  Reliability if an item is dropped:
##    raw_alpha std.alpha G6(smc) average_r S/N alpha se var.r med.r
## H1      0.77      0.75    0.77      0.30 3.0    0.022 0.036  0.23
## H2      0.80      0.78    0.79      0.33 3.5    0.019 0.036  0.31
## H3      0.79      0.77    0.78      0.33 3.4    0.019 0.037  0.24
## H4      0.75      0.74    0.76      0.29 2.9    0.023 0.030  0.24
## H5      0.75      0.74    0.75      0.29 2.8    0.024 0.023  0.24
## H6      0.81      0.80    0.81      0.36 4.0    0.018 0.027  0.37
## H7      0.75      0.74    0.75      0.29 2.9    0.024 0.021  0.24
## H8      0.75      0.74    0.75      0.29 2.8    0.024 0.028  0.23
##
##  Item statistics
##      n raw.r std.r r.cor r.drop mean   sd
## H1 229  0.66  0.66  0.59   0.53  3.1 0.85
## H2 229  0.47  0.53  0.42   0.33  3.4 0.70
## H3 229  0.50  0.56  0.47   0.37  3.4 0.68
## H4 229  0.73  0.71  0.68   0.62  2.9 0.89
## H5 229  0.77  0.73  0.71   0.65  2.7 0.99
## H6 229  0.34  0.41  0.26   0.20  3.4 0.66
## H7 229  0.77  0.72  0.70   0.64  2.6 1.10
## H8 229  0.76  0.72  0.68   0.63  2.8 1.02
##
## Non missing response frequency for each item
##       1    2    3    4 miss
## H1 0.09 0.07 0.53 0.31    0
## H2 0.03 0.05 0.45 0.48    0
## H3 0.03 0.02 0.44 0.52    0

## H4 0.07 0.23 0.43 0.27    0
## H5 0.15 0.25 0.37 0.23    0
## H6 0.01 0.06 0.46 0.46    0
## H7 0.22 0.18 0.34 0.26    0
## H8 0.17 0.11 0.45 0.28    0
```

Analisis Regresi Logistik Biner

```
library(readxl)
data5 <- read_excel("dataskripsi3.xlsx")
str(data5)
```

```
## tibble [229 × 9] (S3: tbl_df/tbl/data.frame)
##  $ X1: num [1:229] 1 0 0 1 0 1 1 1 1 1 ...
##  $ X2: num [1:229] 23 20 22 23 27 28 27 29 27 28 ...
##  $ X3: num [1:229] 0 1 0 1 0 0 0 1 0 0 ...
##  $ X4: num [1:229] 0 0 1 1 0 1 0 1 0 1 ...
##  $ X5: num [1:229] 1 2 1 2 2 2 2 2 2 2 ...
##  $ X7: num [1:229] 0 1 0 1 0 0 0 1 1 1 ...
##  $ X8: num [1:229] 0 0 1 1 0 1 0 1 0 0 ...
##  $ X9: num [1:229] 0 1 0 1 0 0 1 1 0 1 ...
##  $ Y : num [1:229] 0 1 0 1 1 1 0 0 1 0 ...
```

Merubah Variabel menjadi Faktor

```
yyy=as.factor(data5$Y)
xxx1=as.factor(data5$X1)
xxx2=(data5$X2)
xxx3=as.factor(data5$X3)
xxx4=as.factor(data5$X4)
xxx5=as.factor(data5$X5)
xxx7=as.factor(data5$X7)
xxx8=as.factor(data5$X8)
xxx9=as.factor(data5$X9)

Data5 = data.frame(yyy,xxx1,xxx2,xxx3,xxx4,xxx5,xxx7,xxx8,xxx9)
str(Data5)
```

```
## 'data.frame':    229 obs. of  9 variables:
##  $ yyy : Factor w/ 2 levels "0","1": 1 2 1 2 2 2 1 1 2 1 ...
##  $ xxx1: Factor w/ 2 levels "0","1": 2 1 1 2 1 2 2 2 2 2 ...
##  $ xxx2: num  23 20 22 23 27 28 27 29 27 28 ...
##  $ xxx3: Factor w/ 2 levels "0","1": 1 2 1 2 1 1 1 2 1 1 ...
##  $ xxx4: Factor w/ 2 levels "0","1": 1 1 2 2 1 2 1 2 1 2 ...
##  $ xxx5: Factor w/ 2 levels "1","2": 1 2 1 2 2 2 2 2 2 2 ...
##  $ xxx7: Factor w/ 2 levels "0","1": 1 2 1 2 1 1 1 2 2 2 ...
##  $ xxx8: Factor w/ 2 levels "0","1": 1 1 2 2 1 2 1 2 1 1 ...
##  $ xxx9: Factor w/ 2 levels "0","1": 1 2 1 2 1 1 2 2 1 2 ...
```

Uji Independensi

```
attach(Data5)
```

```
## The following objects are masked _by_ .GlobalEnv:
##
##     xxx1, xxx2, xxx3, xxx4, xxx5, xxx7, xxx8, xxx9, yyy
```

```
chisq13 = chisq.test(yyy,xxx1)
chisq13
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  yyy and xxx1
## X-squared = 0.020654, df = 1, p-value = 0.8857
```

```
chisq33 = chisq.test(yyy,xxx3)
chisq33
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  yyy and xxx3
## X-squared = 0.013096, df = 1, p-value = 0.9089
```

```
chisq43 = chisq.test(yyy,xxx4)
chisq43
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  yyy and xxx4
## X-squared = 8.3567, df = 1, p-value = 0.003843
```

```
chisq53 = chisq.test(yyy,xxx5)
chisq53
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  yyy and xxx5
## X-squared = 12.838, df = 1, p-value = 0.0003396
```

```
chisq73 = chisq.test(yyy,xxx7)
chisq73
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  yyy and xxx7
## X-squared = 1.6447, df = 1, p-value = 0.1997
```

```
chisq83 = chisq.test(yyy,xxx8)
chisq83
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  yyy and xxx8
## X-squared = 0.26795, df = 1, p-value = 0.6047
```

```
chisq93 = chisq.test(yyy,xxx9)
chisq93
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  yyy and xxx9
## X-squared = 3.131, df = 1, p-value = 0.07682
```

Estimasi Parameter Logit

```
library(pscl)
```

```
## Classes and Methods for R originally developed in the
## Political Science Computational Laboratory
## Department of Political Science
## Stanford University (2002-2015),
## by and under the direction of Simon Jackman.
## hurdle and zeroinfl functions by Achim Zeileis.
```

```
logit13 = glm(yyy~xxx2+xxx4+xxx5+xxx7+xxx9, family=binomial (link = "logit"))
logit13
```

```
##
## Call:  glm(formula = yyy ~ xxx2 + xxx4 + xxx5 + xxx7 + xxx9, family = binomial(link = "log
it"))
##
## Coefficients:
## (Intercept)         xxx2        xxx41        xxx52        xxx71        xxx91
##     1.31246     -0.03352     -0.76952      1.10029      0.40815     -0.62987
##
## Degrees of Freedom: 228 Total (i.e. Null);  223 Residual
## Null Deviance:       293.8
## Residual Deviance: 266.1     AIC: 278.1
```

Uji Signifikansi Parameter

```
#pengujian serentak
library(lmtest)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
lrtest(logit13)
```

```
## Likelihood ratio test
##
## Model 1: yyy ~ xxx2 + xxx4 + xxx5 + xxx7 + xxx9
## Model 2: yyy ~ 1
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   6 -133.04
## 2   1 -146.89 -5 27.701  4.163e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
qchisq(0.80,5)
```

```
## [1] 7.289276
```

```
#pengujian parsial
summary(logit13)
```

```
##
## Call:
## glm(formula = yyy ~ xxx2 + xxx4 + xxx5 + xxx7 + xxx9, family = binomial(link = "logit"))
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.31246    0.83903   1.564 0.117758
## xxx2        -0.03352    0.02970  -1.128 0.259130
## xxx41       -0.76952    0.29922  -2.572 0.010119 *
## xxx52        1.10029    0.30702   3.584 0.000339 ***
## xxx71        0.40815    0.31468   1.297 0.194621
## xxx91       -0.62907    0.30450  -2.066 0.038838 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 293.78  on 228  degrees of freedom
## Residual deviance: 266.08  on 223  degrees of freedom
## AIC: 278.08
##
## Number of Fisher Scoring iterations: 4
```

Estimasi Parameter Logit 2

```
logit23 = glm(yyy~xxx4+xxx5+xxx7+xxx9, family=binomial (link = "logit"))
logit23
```

```
##
## Call:  glm(formula = yyy ~ xxx4 + xxx5 + xxx7 + xxx9, family = binomial(link = "logit"))
##
## Coefficients:
## (Intercept)        xxx41        xxx52        xxx71        xxx91
##      0.4322      -0.7472       1.0519       0.4273      -0.6082
##
## Degrees of Freedom: 228 Total (i.e. Null);  224 Residual
## Null Deviance:       293.8
## Residual Deviance: 267.4     AIC: 277.4
```

```
qchisq(0.80,5)
```

```
## [1] 7.289276
```

```
#pengujian serentak
library(lmtest)
lrtest(logit23)
```

```
## Likelihood ratio test
##
## Model 1: yyy ~ xxx4 + xxx5 + xxx7 + xxx9
## Model 2: yyy ~ 1
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   5 -133.68
## 2   1 -146.89 -4 26.411  2.614e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
qchisq(0.80,5)
```

```
## [1] 7.289276
```

```
#pengujian parsial
summary(logit23)
```

```
##
## Call:
## glm(formula = yyy ~ xxx4 + xxx5 + xxx7 + xxx9, family = binomial(link = "logit"))
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.4322     0.3062   1.411 0.158099
## xxx41        -0.7472     0.2971  -2.515 0.011908 *
## xxx52         1.0519     0.3021   3.482 0.000498 ***
## xxx71         0.4273     0.3135   1.363 0.172813
## xxx91        -0.6082     0.3029  -2.008 0.044656 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 293.78  on 228  degrees of freedom
## Residual deviance: 267.37  on 224  degrees of freedom
## AIC: 277.37
##
## Number of Fisher Scoring iterations: 4
```

Uji Kesesuaian Model

```
library(ResourceSelection)
```

```
## ResourceSelection 0.3-6   2023-06-27
```

```
hoslem.test(logit23$y,fitted(logit23))
```

```
## Warning in hoslem.test(logit23$y, fitted(logit23)): The data did not allow for
## the requested number of bins.
```

```
##
##  Hosmer and Lemeshow goodness of fit (GOF) test
##
## data:  logit23$y, fitted(logit23)
## X-squared = 2.7846, df = 7, p-value = 0.9042
```

Odds Ratio

```
beta23 = coef(logit23)
OR23 = exp(beta23)
cbind(beta23,OR23)
```

```
##                 beta23      OR23
## (Intercept)  0.4321933 1.5406328
## xxx41       -0.7472359 0.4736740
## xxx52        1.0519135 2.8631245
## xxx71        0.4273039 1.5331185
## xxx91       -0.6081965 0.5443317
```

Ketepatan Klasifikasi

```
library(caret)
```

```
## Loading required package: ggplot2
```

```
## Loading required package: lattice
```

```
predict23 = predict(logit23,type = "response")
fit23 = ifelse(predict23> 0.5, "1","0")
fit23
```

```
##   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20 
## "1" "1" "0" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "0" "1" "1" "1" "1" "1" 
##  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40 
## "1" "1" "1" "0" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" 
##  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60 
## "1" "1" "0" "1" "1" "1" "0" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "0" "1" 
##  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80 
## "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "0" "1" "1" "1" "1" "0" "1" "1" "1" 
##  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99 100 
## "1" "1" "1" "1" "1" "0" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" 
## 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 
## "0" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "0" "1" "1" "1" "1" "1" 
## 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 
## "1" "1" "1" "1" "0" "1" "1" "1" "1" "1" "0" "1" "1" "0" "1" "1" "0" "1" "1" "1" 
## 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 
## "0" "0" "0" "0" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" 
## 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 
## "1" "1" "1" "1" "1" "0" "1" "1" "1" "0" "1" "1" "1" "1" "0" "1" "1" "0" "1" "1" 
## 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 
## "0" "1" "1" "0" "1" "0" "1" "0" "0" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" "1" 
## 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 
## "0" "1" "1" "1" "0" "1" "1" "1" "1" "0" "1" "1" "0" "1" "0" "1" "0" "0" "0" "1" 
## 221 222 223 224 225 226 227 228 229 
## "0" "1" "1" "1" "1" "1" "0" "1" "0"
```

```
tab23 = table(data5$Y,fit23)
tab23
```

```
##    fit23
##       0   1
##   0  22  56
##   1  17 134
```

```
confusionMatrix(tab23, positive="1")
```

```
## Confusion Matrix and Statistics
##
##     fit23
##        0   1
##    0  22  56
##    1  17 134
##
##                Accuracy : 0.6812
##                  95% CI : (0.6166, 0.7411)
##     No Information Rate : 0.8297
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.1928
##
##  Mcnemar's Test P-Value : 8.685e-06
##
##             Sensitivity : 0.7053
##             Specificity : 0.5641
##          Pos Pred Value : 0.8874
##          Neg Pred Value : 0.2821
##              Prevalence : 0.8297
##          Detection Rate : 0.5852
##    Detection Prevalence : 0.6594
##       Balanced Accuracy : 0.6347
##
##        'Positive' Class : 1
##
```

```
prop3 = sum(data5$Y)/nrow(data5)
prop3
```

```
## [1] 0.6593886
```

```
fit3 = glm(data5$Y ~ data5$X1 + data5$X5 + data5$X7 +data5$X9, family = binomial, data = data
5)
predicted3 = as.numeric(fitted(fit3) > prop3)
xtabs(~ data5$Y +predicted3)
```

```
##        predicted3
## data5$Y  0  1
##       0 54 24
##       1 62 89
```