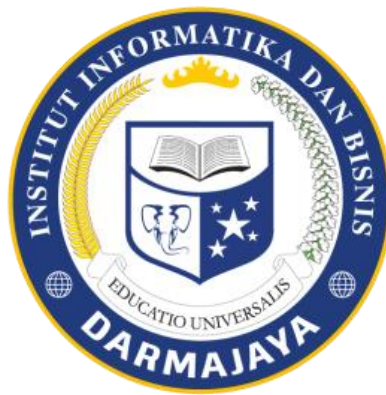


UTS
TOOLS AND TECHNIQUES FOR COMPUTATIONAL ANALYSIS

STUDI KASUS UNTUK ANALISIS CREDIT SCORECARD



Disusun oleh :

Nama :

NPM :

Dosen :

Joko Triloka, S.Kom., M.T., Ph.D

PROGRAM STUDI MAGISTER TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
INSTITUT INFORMATIKA DAN BISNIS DARMAJAYA
TAHUN 2024

KATA PENGANTAR

Assalamu'alaikum Wr. Wb.

Dengan mengucapkan Alhamdulillah segala puji dan syukur penulis panjatkan atas kehadiran Allah SWT, karena berkat rahmat dan hidayah-Nya penulis bisa menyelesaikan UTS yang berisi tentang **STUDI KASUS UNTUK ANALISIS CREDIT SCORECARD** yang terdapat pada Matlab Live Editor dengan alamat website <https://www.mathworks.com/help/finance/case-study-for-a-credit-scorecard-analysis.html>. Studi kasus ini mengilustrasikan langkah-langkah dalam membangun model "credit scorecard" untuk penilaian kredit. Berikut tahapan – tahapan sekaligus halaman dari setiap tahapan untuk mempermudah membaca isi UTS ini :

1. BUAT OBJEK CREDITSCORECARD.....	1
2. a. OTOMATISASI BIN DATA	14
2. b. PENYEMPURNAAN BIN SECARA MANUAL	16
3. SESUAIKAN MODEL REGRESI LOGISTIK	27
4. TINJAU DAN FORMAT POIN SCORECARD	29
5. SCORE DATA	36
6. HITUNG PROBABILITAS GAGAL BAYAR	38

Penulis berharap semoga UTS ini dapat bermanfaat bagi pembaca dan dapat dijadikan referensi. Kebenaran datangnya dari Allah SWT dan kesalahan datangnya dari diri penulis. Semoga Allah SWT senantiasa melimpahkan Rahmat dan Ridho-Nya kepada kita semua.

Wassalamu'alaikum Wr. Wb.

Bandar Lampung, November 2024

Penulis,

STUDI KASUS UNTUK ANALISIS CREDIT SCORECARD

Studi kasus ini mengilustrasikan langkah-langkah dalam membangun model "credit scorecard" untuk penilaian kredit. Tahapan pertama adalah menciptakan objek credit scorecard, yaitu perangkat yang memungkinkan pemodelan risiko kredit berdasarkan data yang telah disusun (data binned). Setelah data binned dikembangkan, kita bisa menampilkan dan memvisualisasikannya untuk memahami distribusi variabel-variabel yang relevan dengan risiko kredit. Langkah berikutnya adalah menyesuaikan model menggunakan regresi logistik, yang merupakan metode statistik untuk menghubungkan variabel independen dengan kemungkinan gagal bayar. Setelah model disesuaikan, skor dapat dihitung berdasarkan model tersebut, memungkinkan penilaian risiko kredit dalam bentuk probabilitas gagal bayar (default probability). Terakhir, validasi model dilakukan dengan menggunakan tiga metrik evaluasi yang berbeda, memastikan bahwa model kartu skor kredit ini akurat dan dapat diandalkan untuk menilai risiko kredit debitur. Penjelasan ini memberikan panduan tentang proses pembuatan model credit scorecard dari awal hingga evaluasi performa model.

1. BUAT OBJEK CREDITSCORECARD

Hal pertama yang dilakukan yaitu gunakan file CreditCardData.mat (dataset dari Refaat, 2011) untuk memuat data. Berikut perintah yang harus dilakukan :

```
load CreditCardData
head(data)
```

Perintah diatas memiliki arti sebagai berikut :

- load CreditCardData : Perintah `load` digunakan untuk memuat data yang disimpan di dalam file MATLAB (biasanya dengan ekstensi `.mat`). Dalam kasus ini, MATLAB akan mencari file bernama `CreditCardData.mat` di direktori kerja dan memuat variabel-variabel yang ada di dalam file tersebut ke dalam workspace.
- head(data) : Fungsi `head` digunakan untuk menampilkan beberapa baris pertama dari variabel `data`, yang memudahkan kita untuk melihat struktur dan isi dari data yang baru saja dimuat.

Berikut hasil tabel yang didapat dari perintah diatas :

CustID	CustAge	TmAtAddress	ResStatus	EmpStatus	CustIncome	TmWBank	OtherCC	AMBalance	UtilRate	status
1	53	62	Tenant	Unknown	50000	55	Yes	1055.9	0.22	0
2	61	22	Home Owner	Employed	52000	25	Yes	1161.6	0.24	0
3	47	30	Tenant	Employed	37000	61	No	877.23	0.29	0
4	50	75	Home Owner	Employed	53000	20	Yes	157.37	0.08	0
5	68	56	Home Owner	Employed	53000	14	Yes	561.84	0.11	0
6	65	13	Home Owner	Employed	48000	59	Yes	968.18	0.15	0
7	34	32	Home Owner	Unknown	32000	26	Yes	717.82	0.02	1
8	50	57	Other	Employed	51000	33	No	3041.2	0.13	0

Berikut penjelasan detail mengenai kolom-kolom yang terdapat dalam data yang ditampilkan :

Berikut adalah parafrase dari penjelasan kolom-kolom data tersebut:

- CustID : Merupakan kode unik untuk setiap pelanggan, yang berupa angka sebagai identifikasi eksklusif setiap baris data.
- CustAge : Menyatakan umur pelanggan dalam bentuk angka, menunjukkan usia mereka dalam tahun.
- TmAtAddress : Menandakan durasi pelanggan tinggal di alamat sekarang, diukur dalam tahun. Misalnya, nilai 75 berarti pelanggan telah tinggal di lokasi tersebut selama 75 tahun.
- ResStatus : Menunjukkan status tempat tinggal pelanggan, apakah sebagai Tenant (penyewa), Home Owner (pemilik rumah), atau kategori Other (lain-lain).
- EmpStatus : Status pekerjaan pelanggan, seperti Employed (bekerja) atau Unknown (tidak diketahui). Ini membantu mengevaluasi kestabilan penghasilan pelanggan.
- CustIncome : Pendapatan tahunan pelanggan dalam mata uang tertentu, yang menjadi indikator kemampuan finansial mereka.
- TmWBank : Lama waktu pelanggan telah menjadi nasabah bank, dalam tahun. Misalnya, nilai 14 berarti pelanggan telah menjadi nasabah selama 14 tahun.
- OtherCC : Menunjukkan apakah pelanggan memiliki kartu kredit lain, dengan nilai Yes (ya) atau No (tidak), yang mengindikasikan keterlibatan mereka dengan kartu kredit dari lembaga lain.
- AMBalance : Saldo rata-rata akun pelanggan dalam satuan tertentu, yang menggambarkan rata-rata saldo yang dimiliki pelanggan.
- UtilRate : Rasio pemakaian kredit, yang menunjukkan persentase penggunaan limit kredit pelanggan. Misalnya, nilai 0.24 berarti penggunaan sebesar 24% dari limit kredit.

- Status : Menunjukkan kondisi tertentu pelanggan, yang mungkin mencerminkan status terkait risiko kredit atau riwayat pembayaran, dengan 0 atau 1 sebagai indikator tanpa masalah atau adanya risiko tertentu.

Data ini secara keseluruhan bisa digunakan untuk menganalisis profil pelanggan, kemampuan finansial, stabilitas pendapatan, dan kecenderungan mereka dalam menggunakan kredit atau produk keuangan lainnya.

```
sc = creditscorecard(data, 'IDVar', 'CustID')
```

Perintah diatas digunakan untuk membuat model Credit Scorecard dari data kredit pelanggan. Fungsi `creditscorecard` di MATLAB membuat objek Credit Scorecard yang membantu menganalisis risiko kredit berdasarkan atribut pelanggan, seperti usia, status pekerjaan, dan lama menjadi nasabah, guna menilai risiko gagal bayar atau kelayakan kredit. Dalam data ini, kolom `CustID` berfungsi sebagai identitas unik untuk setiap pelanggan, yang ditandai menggunakan parameter `IDVar`, sehingga memudahkan pelacakan skor kredit tiap pelanggan. Variabel `sc` menyimpan objek scorecard yang dihasilkan, yang nantinya bisa digunakan untuk menghitung skor kredit dan analisis lanjutan. Langkah ini adalah bagian awal dalam membangun model risiko kredit yang efektif untuk pengelolaan risiko terkait pelanggan. Berikut hasil dari perintah tersebut :

```
sc =
  creditscorecard with properties:
      GoodLabel: 0
      ResponseVar: 'status'
      WeightsVar: ''
      VarNames: {'CustID' 'CustAge' 'TmAtAddress' 'ResStatus'
'EmpStatus' 'CustIncome' 'TmWBank' 'OtherCC' 'AMBalance' 'UtilRate' 'status'}
      NumericPredictors: {'CustAge' 'TmAtAddress' 'CustIncome' 'TmWBank'
'AMBalance' 'UtilRate'}
      CategoricalPredictors: {'ResStatus' 'EmpStatus' 'OtherCC'}
      BinMissingData: 0
      IDVar: 'CustID'
      PredictorVars: {'CustAge' 'TmAtAddress' 'ResStatus' 'EmpStatus'
'CustIncome' 'TmWBank' 'OtherCC' 'AMBalance' 'UtilRate'}
      Data: [1200x11 table]
```

Kode di atas menunjukkan detail objek creditscorecard yang disimpan dalam variabel sc.

- GoodLabel menunjukkan label respons positif, di sini bernilai 0.

- ResponseVar merujuk pada variabel target atau respons, yaitu `status`.
- WeightsVar tidak digunakan (kosong), berarti tidak ada bobot khusus untuk observasi.
- VarNames adalah daftar semua variabel dalam dataset, seperti `CustID`, `CustAge`, dan lainnya.
- NumericPredictors dan CategoricalPredictors mengelompokkan prediktor berdasarkan tipe: numerik (seperti usia dan pendapatan) dan kategorik (seperti status tempat tinggal).
- BinMissingData diatur ke 0, yang berarti data yang hilang tidak akan diproses khusus (dibin).
- IDVar menunjukkan `CustID` sebagai kolom ID unik.
- PredictorVars adalah daftar variabel prediktor yang digunakan dalam model.
- Data menunjukkan bahwa data berukuran 1200 baris x 11 kolom, disimpan dalam bentuk tabel.

Secara keseluruhan, objek ini mempersiapkan data kredit untuk dianalisis menggunakan scorecard kredit.

```
bininfo(sc, 'ResStatus')
```

Kode ini digunakan untuk memperoleh informasi mengenai binning atau pengelompokan nilai dari variabel `ResStatus` dalam objek `creditscorecard` (`sc`). Fungsi `bininfo` berfungsi untuk melihat bagaimana variabel tertentu dibagi menjadi beberapa bin dalam model scorecard. Proses binning ini mengelompokkan data variabel prediktor agar lebih terstruktur, yang penting dalam analisis skor kredit. Variabel `ResStatus` sendiri menunjukkan status tempat tinggal pelanggan, seperti "Tenant," "Home Owner," dan "Other." Ketika menjalankan `bininfo(sc, 'ResStatus')`, hasilnya akan menampilkan detail seperti jumlah bin yang terbentuk, kategori dalam setiap bin, **WoE** (Weight of Evidence) untuk menilai pengaruh tiap kategori terhadap risiko kredit, serta **IV** (Information Value) yang mengukur kekuatan prediktif variabel ini. Berikut hasil ResStatus yang didapat dari kode tersebut :

```
ans = 4x6 table
```

	Bin	Good	Bad	Odds	WOE	InfoValue
1	'Home Own...	365	177	2.0621	0.0193	1.6820e-04
2	'Tenant'	307	167	1.8383	-0.0956	0.0037
3	'Other'	131	53	2.4717	0.2005	0.0059
4	'Totals'	803	397	2.0227	NaN	0.0098

Berikut penjelasan dari hasil output ``bininfo(sc, 'ResStatus')`` yang dihasilkan diatas :

- Bin : Kolom ini menampilkan kategori dari ``ResStatus``, seperti Home Owner, Tenant, dan Other yang telah dikelompokkan.
- Good : Menunjukkan jumlah pelanggan yang tidak berisiko (dengan GoodLabel bernilai 0) dalam tiap kategori.
- Bad : Menampilkan jumlah pelanggan berisiko dalam setiap kategori.
- Odds : Rasio antara pelanggan "Good" dan "Bad" di setiap bin, misalnya, Home Owner memiliki odds 2.0621, yang berarti ada sekitar dua pelanggan "Good" untuk setiap satu pelanggan "Bad".
- WOE (Weight of Evidence) : Nilai WOE mengindikasikan pengaruh kategori terhadap risiko kredit; WOE positif (seperti 0.2005 pada Other) menunjukkan lebih banyak pelanggan "Good" daripada "Bad," sementara nilai negatif (seperti -0.0956 pada Tenant) menunjukkan sebaliknya.
- InfoValue : Information Value (IV) mengukur kekuatan prediktif variabel; total IV untuk ``ResStatus`` adalah 0.0098, menunjukkan bahwa variabel ini lemah dalam memprediksi risiko kredit.

Kategori Home Owner memiliki odds sekitar 2.0621 dan WOE yang sangat rendah (0.0193), artinya status kepemilikan rumah ini sedikit berkontribusi dalam membedakan risiko kredit. Tenant memiliki odds lebih rendah (1.8383) dan WOE negatif (-0.0956), menunjukkan bahwa penyewa cenderung memiliki risiko yang lebih tinggi. Other memiliki odds tertinggi (2.4717) dan WOE positif (0.2005), artinya pelanggan dalam kategori ini lebih cenderung memiliki risiko rendah. Nilai IV total yang rendah (0.0098) menunjukkan bahwa variabel ``ResStatus`` mungkin kurang kuat sebagai prediktor risiko kredit sendiri, dan perlu digabungkan dengan variabel lainnya untuk analisis yang lebih efektif.

Coba kita ubah variabel 'ResStatus' menjadi 'OtherCC' maka hasil yang didapat sebagai berikut :

```
bininfo(sc, 'OtherCC')
```

```
ans = 3x6 table
```

	Bin	Good	Bad	Odds	WOE	InfoValue
1	'No'	116	75	1.5467	-0.2683	0.0119
2	'Yes'	687	322	2.1335	0.0534	0.0024
3	'Totals'	803	397	2.0227	NaN	0.0143

Kolom kategori (bin) menunjukkan dua jenis kelompok dalam variabel OtherCC : No, yang mewakili pelanggan yang tidak memiliki kartu kredit lain, dan Yes, untuk pelanggan yang memiliki kartu kredit lain. Dalam hal jumlah pelanggan "Good" (tidak berisiko), terlihat bahwa pada kategori No terdapat 116 pelanggan yang tergolong Good. Sementara itu, pada kategori Yes, jumlah pelanggan "Bad" (berisiko) mencapai 322. Odds atau rasio antara pelanggan "Good" dan "Bad" menunjukkan bahwa untuk kategori No, nilai odds sebesar 1.5467. Ini berarti bahwa perbandingan pelanggan "Good" terhadap "Bad" di kelompok ini sekitar 1,55:1. Sebaliknya, pada kategori Yes, odds lebih tinggi, yaitu 2.1335, yang mengindikasikan bahwa pelanggan dengan kartu kredit lain cenderung lebih aman dibandingkan dengan mereka yang tidak memiliki kartu kredit lain.

Nilai Weight of Evidence (WOE) juga memberikan gambaran tambahan mengenai risiko. Pada kategori No, WOE negatif sebesar -0.2683 menunjukkan bahwa pelanggan tanpa kartu kredit lain cenderung lebih berisiko, karena terdapat lebih banyak pelanggan "Bad" dibandingkan "Good" secara proporsional. Sebaliknya, nilai WOE positif pada kategori Yes, sebesar 0.0534, menunjukkan bahwa pelanggan dengan kartu kredit lain cenderung tidak berisiko, dengan lebih banyak pelanggan "Good" dibandingkan "Bad". Information Value (IV) mengukur kekuatan prediktif variabel OtherCC dalam membedakan antara pelanggan "Good" dan "Bad". Nilai IV untuk kategori No adalah 0.0119, sementara untuk kategori Yes adalah 0.0024, dengan total IV sebesar 0.0143. Ini menunjukkan bahwa variabel OtherCC memiliki pengaruh prediktif yang rendah terhadap risiko kredit.

Secara keseluruhan, pelanggan yang memiliki kartu kredit lain (kategori Yes) cenderung lebih aman, dengan odds yang lebih tinggi dan nilai WOE positif, yang berarti mereka memiliki lebih

banyak pelanggan "Good" daripada "Bad". Sementara itu, pelanggan yang tidak memiliki kartu kredit lain (kategori No) menunjukkan kecenderungan lebih berisiko, dengan odds yang lebih rendah dan nilai WOE negatif. Meskipun demikian, variabel `OtherCC` secara keseluruhan hanya memberikan sedikit informasi tambahan untuk memprediksi risiko kredit.

Coba kita ubah variabel `OtherCC` menjadi `EmpStatus` maka hasil yang didapat sebagai berikut :

```
bininfo(sc, 'EmpStatus')
```

```
ans = 3x6 table
```

	Bin	Good	Bad	Odds	WOE	InfoValue
1	'Unknown'	396	239	1.6569	-0.1995	0.0217
2	'Employed'	407	158	2.5759	0.2418	0.0263
3	'Totals'	803	397	2.0227	NaN	0.0480

Berikut adalah analisis dari tabel binning untuk variabel `EmpStatus` setelah dilakukan modifikasi:

Tabel ini menunjukkan dua kategori dalam variabel EmpStatus : Unknown, yang menunjukkan status pekerjaan yang tidak diketahui, dan Employed, yang menunjukkan pelanggan dengan pekerjaan tetap. Dalam hal jumlah pelanggan "Good" (tidak berisiko), terlihat bahwa pada kategori Unknown terdapat 396 pelanggan yang tergolong "Good" dan 239 pelanggan yang tergolong "Bad". Untuk kategori Employed, jumlah pelanggan "Good" lebih tinggi, yaitu 407, sedangkan jumlah pelanggan "Bad" adalah 158. Nilai Odds (rasio) menunjukkan bahwa untuk kategori Unknown, odds sebesar 1.6569, yang berarti ada sekitar 1,66 pelanggan "Good" untuk setiap pelanggan "Bad" di kategori ini. Pada kategori Employed, odds lebih tinggi, yaitu 2.5759, yang menunjukkan bahwa pelanggan dengan status pekerjaan Employed cenderung lebih aman daripada mereka yang status pekerjaannya tidak diketahui.

Weight of Evidence (WOE) memberikan informasi lebih lanjut terkait risiko. Kategori Unknown memiliki nilai WOE negatif sebesar -0.1995, yang menunjukkan bahwa pelanggan dengan status pekerjaan yang tidak diketahui cenderung lebih berisiko. Sebaliknya, kategori Employed memiliki WOE positif sebesar 0.2418, menunjukkan bahwa pelanggan dengan status pekerjaan tetap cenderung tidak berisiko. Information Value (IV) untuk setiap kategori juga dihitung untuk

menilai kekuatan prediktif variabel EmpStatus. Nilai IV untuk kategori Unknown adalah 0.0217, sedangkan untuk Employed adalah 0.0263. Total IV sebesar 0.0480 menunjukkan bahwa variabel EmpStatus memiliki pengaruh prediktif yang rendah hingga sedang dalam membedakan antara pelanggan "Good" dan "Bad".

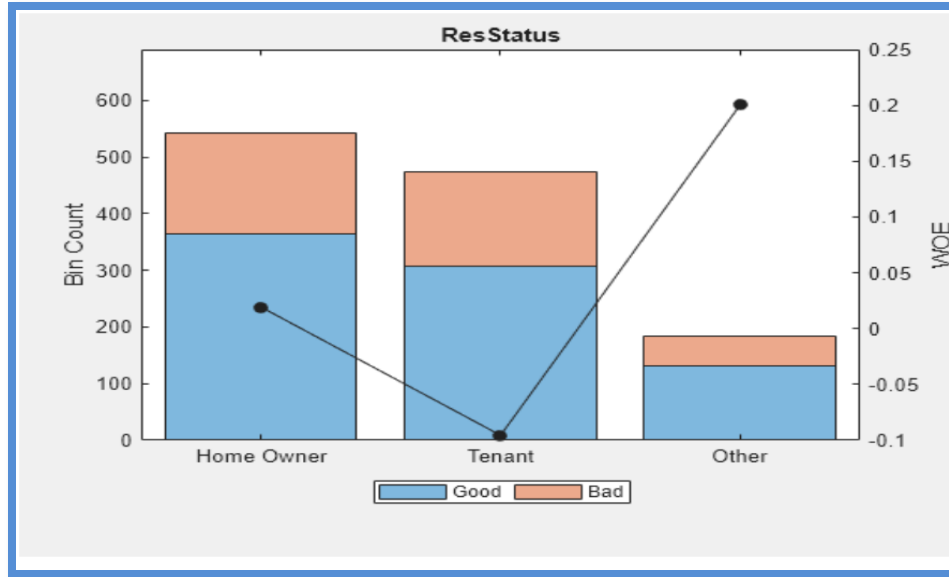
Secara keseluruhan, pelanggan dengan status pekerjaan Employed cenderung lebih aman, dengan odds lebih tinggi dan WOE positif, yang menunjukkan proporsi pelanggan "Good" yang lebih besar. Sebaliknya, pelanggan dengan status pekerjaan yang tidak diketahui (Unknown) menunjukkan kecenderungan lebih berisiko, dengan odds lebih rendah dan WOE negatif. Variabel EmpStatus memberikan kontribusi yang bermanfaat namun terbatas dalam memprediksi risiko kredit pelanggan.

```
plotbins(sc, 'ResStatus')
```

Fungsi plotbins digunakan untuk membuat grafik binning dari variabel tertentu dalam scorecard, seperti `ResStatus`. Grafik ini menunjukkan bagaimana data dalam variabel tersebut dibagi menjadi beberapa kategori (misalnya, "Home Owner," "Tenant," "Other") dan memperlihatkan distribusi pelanggan "Good" dan "Bad" di setiap kategori. Dengan memplot binning untuk `ResStatus` maka dapat :

- Melihat Distribusi : Mengetahui perbandingan jumlah "Good" dan "Bad" di setiap kategori.
- Menginterpretasi WOE : Menilai apakah suatu kategori lebih cenderung memiliki pelanggan "Good" atau "Bad" berdasarkan nilai WOE.
- Analisis Risiko : Mengidentifikasi kategori dengan risiko lebih tinggi atau lebih rendah, yang berguna untuk analisis risiko kredit.

Grafik ini membantu menentukan kategori yang lebih aman atau berisiko. Berikut grafik yang didapat dari plotbins ResStatus :



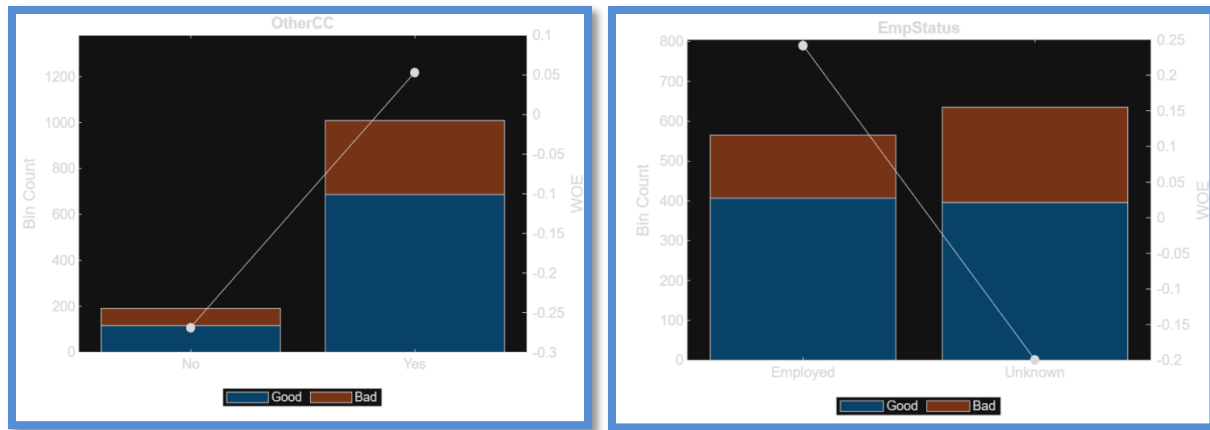
Berikut adalah penjelasan dari grafik binning untuk variabel `ResStatus`:

- Bin Count (Sumbu Y kiri) : Sumbu vertikal kiri menunjukkan jumlah pelanggan (Bin Count) dalam setiap kategori (`Home Owner`, `Tenant`, `Other`). Bar biru mewakili jumlah pelanggan yang tergolong "Good" (tidak berisiko), sementara bar merah muda (merah muda) mewakili pelanggan "Bad" (berisiko). Home Owner memiliki jumlah "Good" yang tinggi dibandingkan "Bad", dengan total lebih dari 500 pelanggan. Tenant memiliki jumlah "Good" yang hampir seimbang dengan "Bad", dengan total sedikit lebih dari 400 pelanggan. Other memiliki jumlah pelanggan paling sedikit, dengan proporsi "Good" yang jauh lebih banyak dibandingkan "Bad".
- WOE (Weight of Evidence, Sumbu Y kanan) : Sumbu vertikal kanan menunjukkan WOE (Weight of Evidence)* untuk setiap kategori dalam variabel `ResStatus`. WOE adalah ukuran yang menunjukkan apakah suatu kategori cenderung memiliki lebih banyak "Good" atau "Bad". Home Owner :WOE mendekati nol, menunjukkan kategori ini netral terhadap risiko. Tenant : WOE negatif, menunjukkan kecenderungan berisiko lebih tinggi. Other : WOE positif tertinggi, menunjukkan kategori ini lebih aman.

Secara keseluruhan, pelanggan dalam kategori Other memiliki kecenderungan paling rendah terhadap risiko karena jumlah pelanggan "Good" lebih banyak dibandingkan "Bad" sehingga mereka dianggap sebagai pelanggan yang lebih aman. Sebaliknya, kategori Tenant menunjukkan kecenderungan risiko yang lebih tinggi, ditandai dengan lebih banyaknya pelanggan "Bad" dibandingkan "Good," yang tercermin dari nilai WOE yang negatif. Sementara itu, kategori

Home Owner memiliki distribusi yang seimbang antara "Good" dan "Bad," dengan nilai WOE mendekati nol, menunjukkan bahwa kategori ini memiliki tingkat risiko yang netral tanpa kecenderungan kuat ke salah satu sisi.

Coba kita lakukan perubahan variabel dengan mengubah variabel ResStatus menjadi OtherCC dan EmpStatus :



Pada grafik OtherCC, terlihat bahwa sebagian besar pelanggan memiliki kartu kredit lain, dengan kategori Yes memiliki jumlah pelanggan "Good" dan "Bad" yang lebih tinggi dibandingkan kategori No. Kategori No menunjukkan jumlah pelanggan "Bad" yang lebih besar daripada "Good," yang mengindikasikan risiko lebih tinggi. Sebaliknya, pada kategori Yes, terdapat lebih banyak pelanggan "Good" dibanding "Bad" yang menunjukkan risiko lebih rendah. Nilai WOE negatif untuk kategori No menegaskan bahwa pelanggan tanpa kartu kredit lain lebih berisiko, sedangkan WOE positif untuk kategori Yes menunjukkan bahwa pelanggan dengan kartu kredit lain cenderung lebih aman.

Pada grafik EmpStatus, terdapat dua kategori yaitu Employed dan Unknown. Kedua kategori ini memiliki jumlah pelanggan yang hampir seimbang. Namun, pada kategori Employed, jumlah pelanggan "Good" lebih besar dibandingkan "Bad" yang menunjukkan risiko lebih rendah bagi mereka yang memiliki pekerjaan tetap. Sebaliknya, pada kategori Unknown, jumlah pelanggan "Bad" lebih banyak daripada "Good" menandakan risiko yang lebih tinggi bagi pelanggan yang status pekerjaannya tidak diketahui. WOE negatif untuk Unknown menunjukkan kecenderungan risiko lebih tinggi, sementara WOE positif untuk Employed mengindikasikan bahwa pelanggan dengan pekerjaan tetap cenderung tidak berisiko.

```

cp = 20000:5000:60000;
sc = modifybins(sc, 'CustIncome', 'CutPoints', cp);
bininfo(sc, 'CustIncome')

```

Kode diatas digunakan untuk melakukan binning atau pengelompokan data pada variabel `CustIncome` dalam objek `creditscorecard` (`sc`) dengan titik potong yang telah ditentukan, lalu melihat hasil pengelompokan tersebut. Pertama, variabel `cp` didefinisikan sebagai `20000:5000:60000`, yang merupakan sintaks MATLAB untuk membuat deretan angka dari 20000 hingga 60000 dengan interval 5000. Hasilnya adalah vektor yang berisi titik-titik potong: `[20000, 25000, 30000, 35000, 40000, 45000, 50000, 55000, 60000]`. Titik-titik ini digunakan untuk membagi nilai `CustIncome` ke dalam beberapa rentang.

Fungsi `modifybins` kemudian diterapkan untuk memodifikasi binning dari variabel `CustIncome` berdasarkan titik potong yang ada di `cp`. Hal ini berarti penghasilan pelanggan akan dikelompokkan sesuai dengan rentang yang telah ditentukan dalam `cp`, sehingga masing-masing rentang ini membentuk satu bin. Parameter `CustIncome` menentukan variabel yang akan dibagi dalam beberapa bin, dan `CutPoints` menunjukkan bahwa binning akan dilakukan sesuai dengan nilai-nilai yang ada di `cp`. Setiap rentang akan menjadi satu kelompok atau bin. Selanjutnya, `bininfo(sc, 'CustIncome')` digunakan untuk menampilkan informasi terkait binning variabel `CustIncome`, seperti jumlah pelanggan "Good" dan "Bad" di setiap bin, serta nilai Odds dan Weight of Evidence (WOE). Informasi ini membantu memahami bagaimana distribusi penghasilan berhubungan dengan risiko kredit. Berikut tabel yang dihasilkan dari kode diatas :

```
ans=11x6 table
```

Bin	Good	Bad	Odds	WOE	InfoValue
{'[-Inf,20000)'} }	3	5	0.6	-1.2152	0.010765
{'[20000,25000)'} }	23	16	1.4375	-0.34151	0.0039819
{'[25000,30000)'} }	38	47	0.80851	-0.91698	0.065166
{'[30000,35000)'} }	131	75	1.7467	-0.14671	0.003782
{'[35000,40000)'} }	193	98	1.9694	-0.026696	0.00017359
{'[40000,45000)'} }	173	76	2.2763	0.11814	0.0028361
{'[45000,50000)'} }	131	47	2.7872	0.32063	0.014348
{'[50000,55000)'} }	82	24	3.4167	0.52425	0.021842
{'[55000,60000)'} }	21	8	2.625	0.26066	0.0015642
{'[60000,Inf]'} }	8	1	8	1.375	0.010235
{'Totals' }	803	397	2.0227	NaN	0.13469

Pada kolom Bin, ditampilkan rentang nilai dari variabel CustIncome berdasarkan titik potong yang telah ditentukan (misalnya, 20000, 25000, dan seterusnya hingga 60000). Setiap rentang menjadi satu kelompok atau bin, seperti {'[-Inf,20000)'} yang mencakup penghasilan di bawah 20000, dan {'[60000,Inf]'} yang mencakup penghasilan di atas 60000. Kolom Good menunjukkan jumlah pelanggan dalam setiap bin yang dikategorikan sebagai "Good" (tidak berisiko), sementara kolom Bad menampilkan jumlah pelanggan yang dianggap "Bad" (berisiko) di setiap bin. Sebagai contoh, dalam rentang {'[35000,40000)'}, terdapat 193 pelanggan yang diklasifikasikan sebagai "Good" dan pada rentang {'[25000,30000)'}, terdapat 47 pelanggan yang diklasifikasikan sebagai "Bad".

Odds menggambarkan rasio jumlah pelanggan "Good" terhadap "Bad" di setiap bin. Misalnya, pada bin {'[30000,35000)'}, odds sebesar 1.7467 menunjukkan rasio sekitar 1.75:1 antara pelanggan "Good" dan "Bad", mengindikasikan peluang lebih tinggi bagi pelanggan dalam bin tersebut untuk dianggap tidak berisiko. WOE (Weight of Evidence) menilai seberapa besar pengaruh setiap kelompok penghasilan terhadap risiko kredit. WOE positif mengindikasikan bahwa lebih banyak pelanggan "Good" dibanding "Bad" dalam suatu kategori, sementara WOE negatif menunjukkan kecenderungan sebaliknya. Sebagai contoh, rentang {'[50000,55000)'}, memiliki WOE positif tertinggi (0.52425), menandakan bahwa pelanggan dalam kelompok penghasilan ini lebih cenderung tidak berisiko.

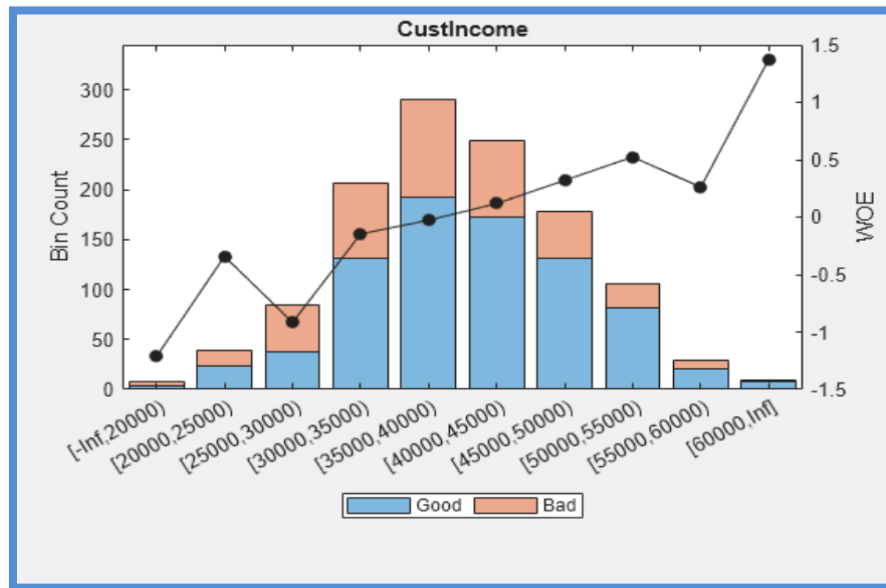
Information Value (IV) mengukur kekuatan prediktif variabel CustIncome dalam membedakan antara pelanggan "Good" dan "Bad". Nilai IV yang lebih tinggi menunjukkan kontribusi yang lebih besar. Dalam hal ini, IV total untuk CustIncome adalah 0.13469, yang menunjukkan kekuatan prediktif yang rendah hingga sedang.

Secara umum, rentang penghasilan yang lebih tinggi memiliki nilai WOE positif, menunjukkan bahwa pelanggan berpenghasilan tinggi cenderung memiliki risiko lebih rendah. Sebaliknya, rentang penghasilan rendah memiliki nilai WOE negatif, menunjukkan kecenderungan risiko lebih tinggi. Dari tabel ini, pelanggan dengan penghasilan antara {'[50000,55000)'}, dan {'[55000,60000)'}, memiliki odds tertinggi dan WOE positif, menjadikan mereka kelompok dengan risiko terendah.

Untuk mempermudah dalam melihat dan memahami data yang telah diklasifikasikan, dapat digunakan fungsi `plotbins()`. Berikut adalah kode untuk memplot data bin yang telah diklasifikasikan pada tahap sebelumnya :

```
plotbins(sc, 'CustIncome')
```

Kode diatas digunakan untuk membuat plot binning dari variabel `CustIncome` dalam objek `creditscorecard` (`sc`). Fungsi ini menghasilkan grafik yang memvisualisasikan distribusi pelanggan "Good" dan "Bad" di setiap bin atau rentang penghasilan (`CustIncome`) yang telah ditentukan. Berikut grafik yang dihasilkan :



Pada sumbu Y kiri, Bin Count menunjukkan jumlah pelanggan di setiap rentang penghasilan. Bar biru mewakili pelanggan "Good" (tidak berisiko), sementara bar merah muda menunjukkan pelanggan "Bad" (berisiko). Rentang penghasilan `40000-45000` dan `45000-50000` memiliki jumlah pelanggan tertinggi dengan banyak kombinasi "Good" dan "Bad," sedangkan rentang di bawah `20000` dan di atas `60000` memiliki jumlah pelanggan yang sangat sedikit.

Sumbu Y kanan menampilkan Weight of Evidence (WOE) untuk setiap bin, yang digambarkan sebagai titik hitam yang terhubung oleh garis. WOE negatif muncul pada rentang penghasilan rendah (di bawah `35000`), menunjukkan kecenderungan risiko lebih tinggi di kelompok ini, dengan lebih banyak pelanggan "Bad" daripada "Good." Di sisi lain, WOE positif muncul pada rentang penghasilan yang lebih tinggi, terutama setelah `45000`, mengindikasikan bahwa pelanggan di kelompok penghasilan ini cenderung tidak berisiko (lebih banyak "Good" dibandingkan "Bad").

Interpretasi Berdasarkan Rentang Penghasilan menunjukkan bahwa:

- Penghasilan Rendah (di bawah `35000`) memiliki WOE negatif, menandakan risiko tinggi bagi pelanggan dalam kelompok ini.
- Penghasilan Menengah (`35000` hingga `50000`) memiliki WOE yang mendekati nol hingga sedikit positif, menunjukkan bahwa kelompok ini relatif netral atau memiliki risiko rendah.
- Penghasilan Tinggi (di atas `50000`) memiliki WOE positif yang meningkat, terutama pada rentang `55000-60000` dan lebih tinggi, menunjukkan bahwa pelanggan dengan penghasilan tinggi umumnya lebih aman atau tidak berisiko.

Kesimpulan dari grafik ini adalah bahwa penghasilan lebih tinggi cenderung berkaitan dengan risiko kredit yang lebih rendah, sementara penghasilan rendah menunjukkan kecenderungan risiko yang lebih tinggi. Grafik ini sangat berguna untuk mengidentifikasi pola risiko kredit berdasarkan tingkat penghasilan, yang dapat mendukung penentuan strategi atau kebijakan kredit, misalnya, dengan menerapkan persyaratan yang lebih ketat bagi kelompok penghasilan rendah.

2. a. OTOMATISASI BIN DATA

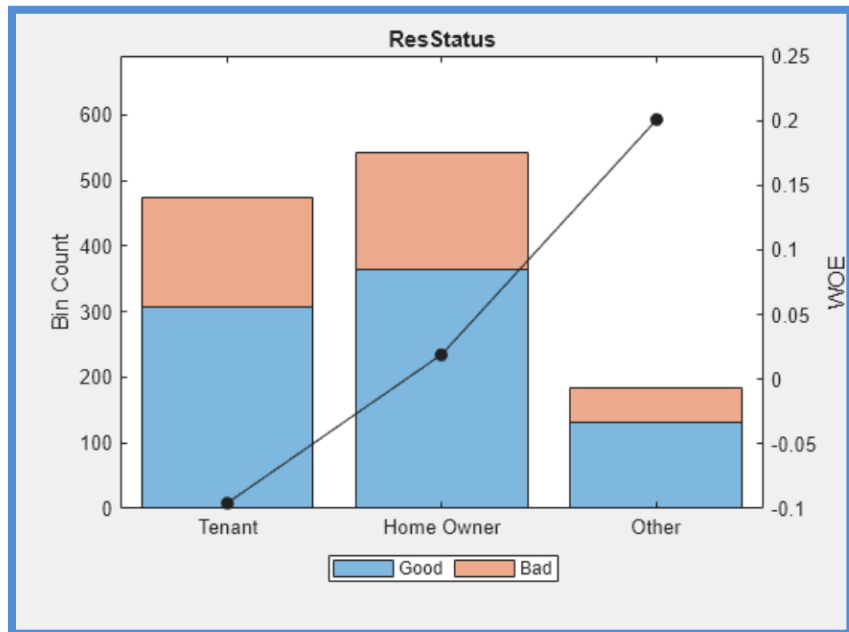
Gunakan fungsi pengelompokan otomatis untuk mengelompokkan setiap variabel prediktor secara otomatis, dengan algoritme bawaan 'Monoton' dan opsi default.

```
sc = autobinning(sc);
```

Perintah `sc = autobinning(sc);` digunakan untuk melakukan pengelompokan otomatis pada data dalam objek `sc`. Fungsi ini akan secara otomatis membagi variabel prediktor ke dalam beberapa kelompok (bin) menggunakan algoritme default, tanpa perlu menentukan titik potong secara manual. Setelah melakukan pengelompokan otomatis, setiap kotak prediktor perlu diperiksa menggunakan fungsi `bininfo` dan `plotbins` untuk disempurnakan. Tren monotonik, yang idealnya linier dalam Weight of Evidence (WOE), sangat diinginkan dalam kartu skor kredit karena dapat diterjemahkan menjadi poin linier untuk prediktor tertentu. Tren WOE ini dapat divisualisasikan dengan menggunakan `plotbins`.

```
Predictor = 'ResStatus';  
plotbins(sc, Predictor)
```

Perintah `plotbins(sc, Predictor)` digunakan untuk memvisualisasikan hasil pengelompokan data pada variabel prediktor yang ditentukan, dalam hal ini variabel `ResStatus`, dan menampilkan distribusi pelanggan "Good" dan "Bad" di setiap kategori serta visualisasi Weight of Evidence (WOE) untuk kategori tersebut. Berikut hasil visualisasinya :



Sumbu Y kiri pada grafik menunjukkan jumlah pelanggan dalam setiap kategori `ResStatus`, yaitu `Tenant`, `Home Owner`, dan `Other`. Kategori `Tenant` dan `Home Owner` memiliki jumlah pelanggan lebih banyak dibandingkan kategori `Other`. Bar biru mewakili pelanggan yang dianggap "Good" (tidak berisiko), sementara bar merah muda menunjukkan pelanggan yang "Bad" (berisiko). Dalam kategori `Tenant`, jumlah pelanggan "Bad" lebih banyak daripada "Good", sedangkan pada kategori `Home Owner`, jumlah pelanggan "Good" lebih banyak daripada "Bad". Kategori `Other` memiliki jumlah pelanggan paling sedikit, namun dengan proporsi "Good" yang lebih banyak dibanding "Bad".

Pada sumbu Y kanan, nilai WOE (Weight of Evidence) untuk setiap kategori menunjukkan kecenderungan risiko berdasarkan status tempat tinggal. Kategori `Tenant` memiliki nilai WOE negatif, yang mengindikasikan bahwa pelanggan dalam kategori ini lebih berisiko karena lebih banyak pelanggan "Bad" dibandingkan "Good". Kategori `Home Owner` memiliki nilai WOE yang mendekati nol, yang menunjukkan bahwa distribusi pelanggan "Good" dan "Bad" lebih seimbang. Sedangkan kategori `Other` memiliki nilai WOE positif yang lebih tinggi, yang

menunjukkan bahwa pelanggan dalam kategori ini lebih aman dengan lebih banyak pelanggan "Good" dibandingkan "Bad".

Secara keseluruhan, pelanggan dalam kategori `Other` cenderung lebih aman, sedangkan `Home Owner` menunjukkan keseimbangan antara risiko rendah dan tinggi. Kategori `Tenant` menunjukkan kecenderungan risiko yang lebih tinggi. Grafik ini membantu dalam mengidentifikasi pola risiko kredit berdasarkan status tempat tinggal, yang dapat menjadi dasar dalam pengambilan keputusan persetujuan kredit.

2. b. PENYEMPURNAAN BIN SECARA MANUAL

Untuk memodifikasi bin secara manual, pertama gunakan fungsi `bininfo` dengan dua output, di mana output kedua berisi aturan binning. Setelah itu, Anda bisa mengubah aturan binning tersebut menggunakan output kedua dari `bininfo`. Selanjutnya, perbarui aturan bin dengan menggunakan `modifybins` dan cek kembali hasil perubahan dengan `plotbins` atau `bininfo`. Sebagai contoh, berdasarkan plot untuk `CustAge` di Langkah 2a, bin 1 dan 2 serta bin 5 dan 6 memiliki WOE yang serupa, sehingga dapat digabungkan menggunakan langkah-langkah di atas.

```
Predictor = 'CustAge';
[bi,cp] = bininfo(sc,Predictor)
```

Perintah di atas digunakan untuk mendapatkan informasi tentang binning pada variabel `CustAge` dalam objek `sc`. Variabel `bi` berisi detail mengenai binning, seperti jumlah pelanggan dan nilai WOE untuk setiap bin, sementara `cp` berisi aturan atau titik potong yang digunakan untuk membagi data `CustAge` ke dalam bin. Berikut hasil yang didapat :

`bi = 8x6 table`

	Bin	Good	Bad	Odds	WOE	InfoValue
1	'[-Inf,29000]'	53	58	0.9138	-0.7946	0.0636
2	'[29000,330...	74	49	1.5102	-0.2922	0.0091
3	'[33000,350...	68	36	1.8889	-0.0684	4.1042e-04
4	'[35000,400...	193	98	1.9694	-0.0267	1.7359e-04
5	'[40000,420...	68	34	2	-0.0113	1.0819e-05
6	'[42000,470...	164	66	2.4848	0.2058	0.0078
7	'[47000,Inf]'	183	56	3.2679	0.4797	0.0417
8	'Totals'	803	397	2.0227	NaN	0.1228

```

cp = 6x1
      29000
      33000
      35000
      40000
      42000
      47000

```

Hasil yang ditampilkan memberikan informasi mengenai binning variabel `CustAge` yang telah dikelompokkan berdasarkan titik potong yang ditentukan. Berikut adalah penjelasan untuk setiap kolom:

- **Bin** : Menunjukkan rentang penghasilan yang digunakan untuk mengelompokkan data. Setiap rentang tersebut menjadi satu bin atau kelompok. Sebagai contoh, `['-Inf,29000')` mencakup pelanggan dengan penghasilan di bawah 29000, sementara `[47000,Inf)` mencakup pelanggan dengan penghasilan lebih dari 47000.
- **Good** : Menyatakan jumlah pelanggan yang tergolong "Good" (tidak berisiko) dalam setiap bin. Misalnya, pada bin `['-Inf,29000')`, terdapat 53 pelanggan yang dianggap "Good".
- **Bad** : Menyatakan jumlah pelanggan yang tergolong "Bad" (berisiko) dalam setiap bin. Sebagai contoh, pada bin `['-Inf,29000')`, terdapat 58 pelanggan yang dianggap "Bad".
- **Odds** : Menunjukkan rasio antara jumlah pelanggan "Good" dan "Bad" dalam setiap bin. Sebagai contoh, pada bin `['-Inf,29000')`, odds sebesar 0.9138 berarti rasio pelanggan "Good" terhadap "Bad" adalah sekitar 0.91:1.
- **WOE (Weight of Evidence)** : Mengukur pengaruh setiap bin terhadap risiko. Nilai WOE yang positif menunjukkan bahwa bin tersebut lebih banyak memiliki pelanggan "Good" dibandingkan "Bad", sementara WOE negatif menunjukkan lebih banyak pelanggan "Bad". Misalnya, pada bin `['-Inf,29000')`, nilai WOE adalah -0.7946, yang menunjukkan bahwa bin ini lebih berisiko karena lebih banyak pelanggan "Bad".
- **InfoValue** : Mengukur seberapa baik bin tersebut dalam membedakan antara pelanggan "Good" dan "Bad". Nilai IV yang lebih tinggi menunjukkan prediktif yang lebih kuat. Sebagai contoh, pada bin `['-Inf,29000')`, nilai InfoValue adalah 0.0636.
- **Cp** : Menampilkan titik potong yang digunakan untuk membagi data menjadi beberapa bin. Titik potong yang digunakan dalam hal ini adalah 29000, 33000, 35000, 40000, 42000, dan 47000.

```

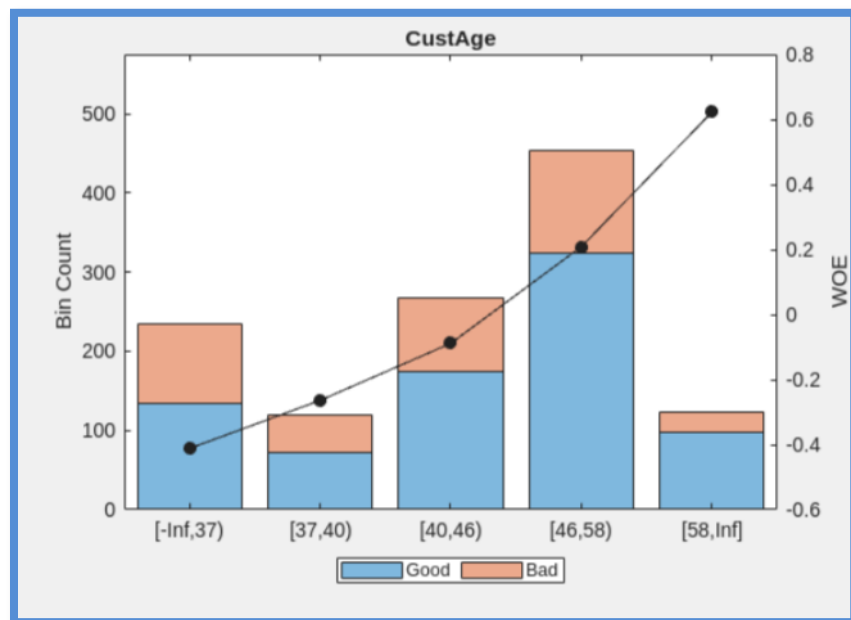
cp([1 5]) = []; % To merge bins 1 and 2, and bins 5 and 6
sc = modifybins(sc, 'CustAge', 'CutPoints', cp);
plotbins(sc, 'CustAge')

```

Kode ini digunakan untuk menggabungkan beberapa bin dan memperbarui pengelompokan data, lalu memvisualisasikan hasilnya. Berikut penjelasannya :

- `cp([1 5]) = [];` : Menghapus elemen pertama dan kelima dari vektor `'cp'`, yang berarti bin pertama dan kedua, serta bin kelima dan keenam, akan digabungkan.
- `'sc = modifybins(sc, 'CustAge', 'CutPoints', cp);'` **: Memperbarui binning untuk variabel `'CustAge'` dengan titik potong yang baru setelah penggabungan.
- `'plotbins(sc, 'CustAge')'` **: Memvisualisasikan hasil pengelompokan yang telah diperbarui, menunjukkan distribusi pelanggan "Good" dan "Bad" setelah penggabungan bin.

Secara keseluruhan, kode ini menggabungkan beberapa bin pada `'CustAge'` dan memvisualisasikan perubahan tersebut untuk memudahkan analisis. Berikut hasil visualisasinya :



Sumbu Y kiri menunjukkan jumlah pelanggan dalam setiap rentang usia (`'CustAge'`). Rentang usia `'[-Inf,37)'` memiliki jumlah pelanggan yang paling sedikit, sedangkan rentang usia `'[46,58)'` mencakup pelanggan terbanyak. Rentang usia `'[37,40)'` dan `'[40,46)'` memiliki jumlah pelanggan yang hampir seimbang, dengan sedikit lebih banyak pelanggan di rentang `'[40,46)'`. Rentang usia `'[58,Inf]'` memiliki jumlah pelanggan yang sedikit lebih rendah dibandingkan rentang usia sebelumnya.

Dalam hal distribusi risiko, bar biru mewakili pelanggan "Good" (tidak berisiko) dan bar merah muda mewakili pelanggan "Bad" (berisiko). Pada setiap bin, jumlah pelanggan "Good" lebih banyak dibandingkan "Bad", meskipun pada beberapa bin, seperti bin $[46,58)$, pelanggan "Bad" lebih menonjol. Misalnya, pada bin $[-Inf,37)$, pelanggan "Good" jauh lebih banyak daripada "Bad". Nilai WOE, yang ditunjukkan sebagai titik hitam yang terhubung dengan garis pada sumbu Y kanan, menunjukkan pengaruh atau kecenderungan risiko di setiap bin. Bin $[-Inf,37)$ memiliki WOE negatif yang signifikan, menunjukkan risiko lebih tinggi, sementara bin $[46,58)$ memiliki WOE positif yang tinggi, menunjukkan risiko lebih rendah. Bin $[37,40)$ dan $[40,46)$ memiliki WOE mendekati nol, yang menunjukkan distribusi yang lebih seimbang antara "Good" dan "Bad". Bin $[58,Inf]$ menunjukkan WOE positif yang lebih rendah, mengindikasikan sedikit pengaruh positif terhadap risiko.

Secara keseluruhan, pelanggan dalam rentang usia $[46,58)$ memiliki risiko lebih rendah, sementara pelanggan dalam rentang usia $[-Inf,37)$ dan $[37,40)$ cenderung lebih berisiko. Grafik ini memberikan gambaran mengenai hubungan antara usia pelanggan dan risiko kredit, dengan WOE yang lebih tinggi menunjukkan lebih banyak pelanggan "Good".

Untuk variabel 'CustIncome', disarankan untuk menggabungkan bin 3, 4, dan 5 karena memiliki nilai WOE yang serupa. Untuk melakukan penggabungan bin-bin tersebut :

```
Predictor = 'CustIncome';  
[bi,cp] = bininfo(sc,Predictor)
```

Perintah diatas digunakan untuk mendapatkan informasi tentang binning pada variabel 'CusIncome' dalam objek 'sc'. Variabel 'bi' berisi detail mengenai binning, seperti jumlah pelanggan dan nilai WOE untuk setiap bin, sementara 'cp' berisi aturan atau titik potong yang digunakan untuk membagi data 'CustIncome' ke dalam bin. Berikut hasil yang didapat :

bi = 8x6 table

	Bin	Good	Bad	Odds	WOE	InfoValue
1	'[-Inf,29000]'	53	58	0.9138	-0.7946	0.0636
2	'[29000,33000]'	74	49	1.5102	-0.2922	0.0091
3	'[33000,35000]'	68	36	1.8889	-0.0684	4.1042e-04
4	'[35000,40000]'	193	98	1.9694	-0.0267	1.7359e-04
5	'[40000,42000]'	68	34	2	-0.0113	1.0819e-05
6	'[42000,47000]'	164	66	2.4848	0.2058	0.0078
7	'[47000,Inf]'	183	56	3.2679	0.4797	0.0417
8	'Totals'	803	397	2.0227	NaN	0.1228

cp = 6x1

```

29000
33000
35000
40000
42000
47000

```

Tabel tersebut menunjukkan pengelompokan pelanggan berdasarkan rentang penghasilan ('CustIncome'). Setiap bin atau rentang penghasilan, seperti '[-Inf,29000)', '[29000,33000)', dan lainnya, mencatat jumlah pelanggan yang dikategorikan sebagai "Good" (tidak berisiko) dan "Bad" (berisiko). Rasio antara pelanggan "Good" dan "Bad" dihitung dalam bentuk odds, sementara Weight of Evidence (WOE) mengukur pengaruh bin terhadap risiko, dengan nilai WOE positif menunjukkan lebih banyak pelanggan "Good". InfoValue menunjukkan efektivitas setiap bin dalam memprediksi risiko, dengan nilai lebih tinggi menandakan prediktif yang lebih baik. Secara keseluruhan, bin dengan penghasilan lebih tinggi, seperti '[42000,47000)' dan '[47000,Inf]', cenderung lebih aman, sementara bin dengan penghasilan lebih rendah, seperti '[-Inf,29000)', lebih berisiko.

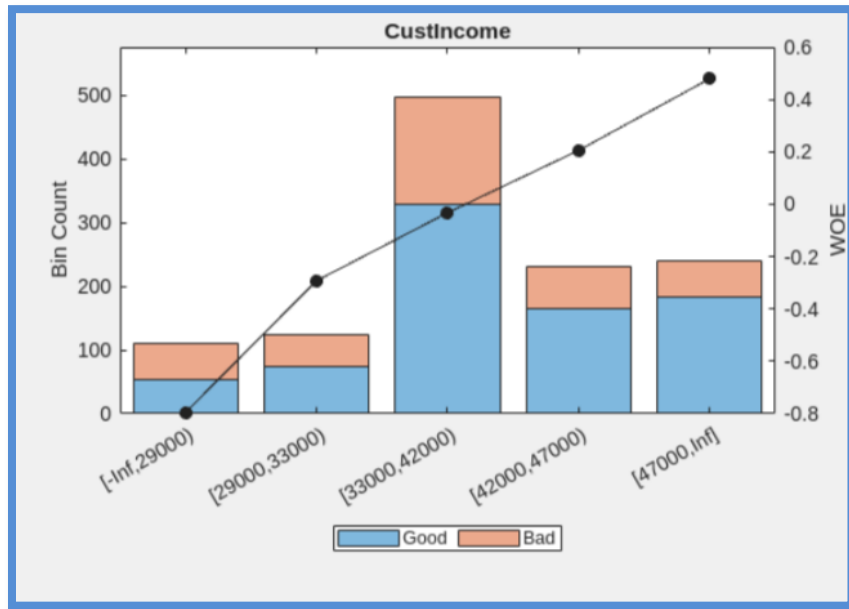
Tabel tersebut juga menampilkan vektor cp, yang berisi titik potong yang digunakan untuk membagi data 'CustIncome' ke dalam bin. Titik potong yang digunakan yaitu 29000, 33000, 35000, 40000, 42000, 47000. Titik potong ini menentukan batas antara setiap bin atau rentang penghasilan yang digunakan dalam pengelompokan pelanggan. Sebagai contoh, rentang '[-Inf,29000)' mencakup pelanggan dengan penghasilan di bawah 29000, dan rentang '[47000,Inf]' mencakup pelanggan dengan penghasilan lebih dari 47000.

```

cp([3 4]) = []; % To merge bins 3, 4, and 5
sc = modifybins(sc, 'CustIncome', 'CutPoints', cp);
plotbins(sc, 'CustIncome')

```

kode diatas menggabungkan beberapa bin pada `CustIncome` dan memvisualisasikan perubahan tersebut untuk memudahkan analisis. Berikut hasil visualisasinya :



Pada sumbu Y kiri, grafik menunjukkan jumlah pelanggan berdasarkan rentang penghasilan (`CustIncome`). Rentang penghasilan $[-\text{Inf}, 29000)$ dan $[29000, 33000)$ memiliki jumlah pelanggan yang lebih sedikit dibandingkan rentang $[33000, 42000)$, yang mencatat jumlah pelanggan terbanyak. Rentang $[42000, 47000)$ dan $[47000, \text{Inf}]$ memiliki jumlah pelanggan yang lebih sedikit, meskipun tetap lebih banyak dibandingkan rentang yang lebih rendah. Grafik ini juga menunjukkan distribusi pelanggan yang "Good" (tidak berisiko) dan "Bad" (berisiko) dengan bar biru untuk pelanggan "Good" dan bar merah muda untuk pelanggan "Bad". Secara umum, lebih banyak pelanggan yang tergolong "Good" dibandingkan "Bad" di setiap bin, meskipun pada bin dengan rentang penghasilan lebih tinggi seperti $[42000, 47000)$ dan $[47000, \text{Inf}]$, proporsi pelanggan "Bad" sedikit lebih besar.

Nilai WOE (Weight of Evidence) yang ditampilkan pada sumbu Y kanan menggambarkan seberapa besar pengaruh setiap bin terhadap risiko. Bin pertama ($[-\text{Inf}, 29000)$) memiliki WOE negatif yang menunjukkan bahwa pelanggan dalam rentang penghasilan ini cenderung lebih berisiko. Bin kedua ($[29000, 33000)$) juga memiliki WOE negatif, meskipun lebih rendah, menandakan risiko yang lebih tinggi. Di sisi lain, bin ketiga ($[33000, 42000)$) dan bin keempat

(`[42000,47000)`) memiliki nilai WOE yang lebih tinggi, menunjukkan bahwa pelanggan dalam rentang penghasilan ini lebih cenderung tidak berisiko. Bin kelima (`[47000,Inf]`) juga memiliki WOE positif yang lebih rendah, namun tetap menunjukkan lebih banyak pelanggan "Good".

Secara keseluruhan, grafik ini menunjukkan bahwa pelanggan dengan penghasilan di rentang `[33000,42000)` memiliki risiko lebih rendah, sementara pelanggan dengan penghasilan di bawah 29000 cenderung lebih berisiko. Penggabungan bin 3, 4, dan 5 yang dilakukan dalam kode menghasilkan distribusi yang lebih baik antara pelanggan "Good" dan "Bad", dengan WOE yang lebih positif di rentang penghasilan yang lebih tinggi.

Berdasarkan plot di atas untuk variabel 'TmWBank', disarankan untuk menggabungkan bin 2 dan 3 karena keduanya memiliki nilai WOE yang hampir sama. Untuk menggabungkan bin-bin tersebut:

```
Predictor = 'TmWBank';
[bi,cp] = bininfo(sc,Predictor)
```

Perintah diatas digunakan untuk mendapatkan informasi tentang binning pada variabel `TmWBank` dalam objek `sc`. Variabel `bi` berisi detail mengenai binning, seperti jumlah pelanggan dan nilai WOE untuk setiap bin, sementara `cp` berisi aturan atau titik potong yang digunakan untuk membagi data `TmWBank` ke dalam bin. Berikut hasil yang didapat :

```
bi = 6x6 table
```

	Bin	Good	Bad	Odds	WOE	InfoValue
1	'[-Inf,12]'	141	90	1.5667	-0.2555	0.0131
2	'[12,23]'	165	93	1.7742	-0.1311	0.0038
3	'[23,45]'	224	125	1.7920	-0.1211	0.0043
4	'[45,71]'	177	67	2.6418	0.2670	0.0138
5	'[71,Inf]'	96	22	4.3636	0.7689	0.0493
6	'Totals'	803	397	2.0227	NaN	0.0843

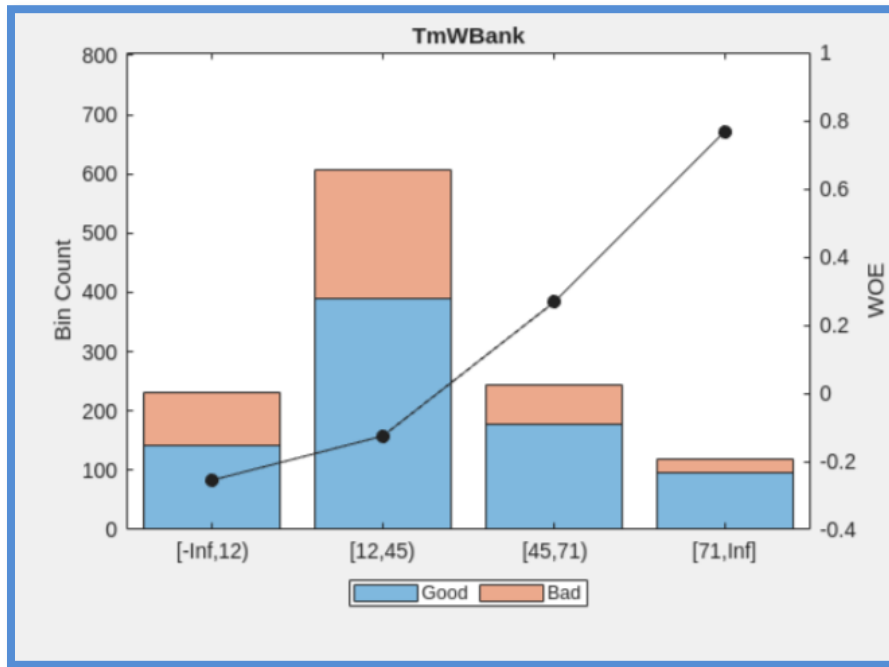
```
cp = 4x1
    12
    23
    45
    71
```

Tabel ini menunjukkan hasil pengelompokan untuk variabel `TmWBank`. Penjelasan setiap kolom adalah sebagai berikut :

- Bin : Menunjukkan rentang nilai dari variabel `TmWBank` yang digunakan untuk mengelompokkan data. Setiap rentang membentuk satu kelompok atau bin. Contohnya, `[-Inf,12)` mencakup nilai `TmWBank` yang lebih kecil dari 12, sementara `[71,Inf)` mencakup nilai lebih besar dari 71.
- Good : Menyajikan jumlah pelanggan yang tergolong "Good" (tidak berisiko) dalam setiap bin. Misalnya, pada bin `[-Inf,12)`, terdapat 141 pelanggan yang dikategorikan sebagai "Good".
- Bad : Menyajikan jumlah pelanggan yang dikategorikan sebagai "Bad" (berisiko) dalam setiap bin. Contohnya, pada bin `[12,23)`, ada 93 pelanggan yang dianggap "Bad".
- Odds : Menunjukkan rasio antara pelanggan "Good" dan "Bad" dalam setiap bin. Sebagai contoh, pada bin `[-Inf,12)`, odds sebesar 1.5667 berarti rasio pelanggan "Good" terhadap "Bad" adalah sekitar 1.57:1.
- WOE (Weight of Evidence) : Mengukur seberapa besar pengaruh bin terhadap risiko. Nilai WOE positif menunjukkan lebih banyak pelanggan "Good", sementara WOE negatif menunjukkan lebih banyak pelanggan "Bad". Misalnya, pada bin `[-Inf,12)`, WOE sebesar -0.2555 menunjukkan kecenderungan risiko yang lebih tinggi.
- InfoValue : Menyatakan kekuatan prediktif setiap bin dalam membedakan antara pelanggan "Good" dan "Bad". Nilai IV yang lebih tinggi menunjukkan prediktif yang lebih baik. Sebagai contoh, bin `[71,Inf)` memiliki InfoValue 0.0493, yang menunjukkan prediktif yang cukup baik.
- cp : Titik potong yang digunakan untuk membagi `TmWBank` menjadi beberapa bin. Titik potong yang digunakan adalah 12, 23, 45, dan 71.

```
cp(2) = []; % To merge bins 2 and 3
sc = modifybins(sc, 'TmWBank', 'CutPoints', cp);
plotbins(sc, 'TmWBank')
```

kode diatas menggabungkan beberapa bin pada `TmWBank` dan memvisualisasikan perubahan tersebut untuk memudahkan analisis. Berikut hasil visualisasinya :



Sumbu Y kiri (Bin Count) menggambarkan jumlah pelanggan dalam setiap rentang nilai `TmWBank`. Bin pertama ($[-\text{Inf}, 12)$) mencatat jumlah pelanggan yang paling sedikit, sedangkan bin kedua ($[12, 45)$) memiliki jumlah pelanggan terbanyak. Bin ketiga ($[45, 71)$) dan keempat ($[71, \text{Inf}]$) mencatat jumlah pelanggan yang lebih sedikit dibandingkan dengan bin kedua. Bar biru mewakili pelanggan yang digolongkan sebagai "Good" (tidak berisiko), sementara bar merah muda mewakili pelanggan yang digolongkan sebagai "Bad" (berisiko). Di setiap bin, lebih banyak pelanggan yang tergolong "Good" dibandingkan dengan "Bad". Namun, pada bin pertama ($[-\text{Inf}, 12)$), proporsi pelanggan "Bad" sedikit lebih tinggi dibandingkan dengan "Good". Pada bin kedua ($[12, 45)$), jumlah pelanggan "Good" lebih banyak daripada "Bad".

Titik hitam yang terhubung dengan garis menunjukkan nilai WOE (Sumbu Y Kanan) untuk setiap bin. Bin pertama ($[-\text{Inf}, 12)$) memiliki WOE negatif, yang menunjukkan bahwa pelanggan dalam rentang ini lebih berisiko, dengan lebih banyak pelanggan "Bad" daripada "Good". Bin kedua ($[12, 45)$) memiliki WOE yang lebih tinggi, menunjukkan bahwa pelanggan dalam rentang ini cenderung lebih aman. Bin ketiga ($[45, 71)$) memiliki WOE yang lebih tinggi lagi, menandakan bahwa pelanggan dengan nilai `TmWBank` lebih tinggi memiliki risiko lebih

rendah. Bin keempat ($[-71, \text{Inf}]$) menunjukkan WOE yang positif, yang menandakan bahwa pelanggan dalam rentang ini sangat sedikit berisiko.

Pelanggan dengan nilai `TmWBank` lebih rendah, seperti pada bin pertama, lebih cenderung berisiko, sementara pelanggan dengan nilai lebih tinggi, seperti pada bin ketiga dan keempat, cenderung lebih aman, dengan lebih banyak pelanggan "Good". Grafik ini mengilustrasikan hubungan antara nilai `TmWBank` dan risiko kredit, dengan WOE yang lebih tinggi menunjukkan lebih banyak pelanggan "Good" dan lebih rendahnya risiko kredit.

Berdasarkan plot di atas, untuk variabel 'AMBalance', disarankan untuk menggabungkan bin 2 dan 3 karena nilai WOE keduanya mirip. Untuk melakukan penggabungan bin tersebut:

```
Predictor = 'AMBalance';  
[bi,cp] = bininfo(sc,Predictor)
```

Perintah di atas digunakan untuk mendapatkan informasi tentang binning pada variabel `AMBalance` dalam objek `sc`. Variabel `bi` berisi detail mengenai binning, seperti jumlah pelanggan dan nilai WOE untuk setiap bin, sementara `cp` berisi aturan atau titik potong yang digunakan untuk membagi data `AMBalance` ke dalam bin. Berikut hasil yang didapat :

```
bi = 5x6 table
```

	Bin	Good	Bad	Odds	WOE	InfoValue
1	'[-Inf,558.88]'	346	134	2.5821	0.2442	0.0228
2	'[558.88,12...	309	171	1.8070	-0.1127	0.0052
3	'[1254.28,1...	76	44	1.7273	-0.1579	0.0026
4	'[1597.44,Inf]'	72	48	1.5000	-0.2990	0.0093
5	'Totals'	803	397	2.0227	NaN	0.0399

```
cp = 3x1  
103 x  
0.5589  
1.2543  
1.5974
```

Tabel ini menunjukkan hasil binning untuk variabel `AMBalance`, dengan setiap rentang nilai membentuk satu bin atau kelompok. Misalnya, rentang $[-\text{Inf}, 558.88]$ mencakup nilai `AMBalance` yang kurang dari 558.88, sementara $[1597.44, \text{Inf}]$ mencakup nilai lebih besar dari 1597.44. Kolom "Good" menunjukkan jumlah pelanggan yang dikategorikan sebagai

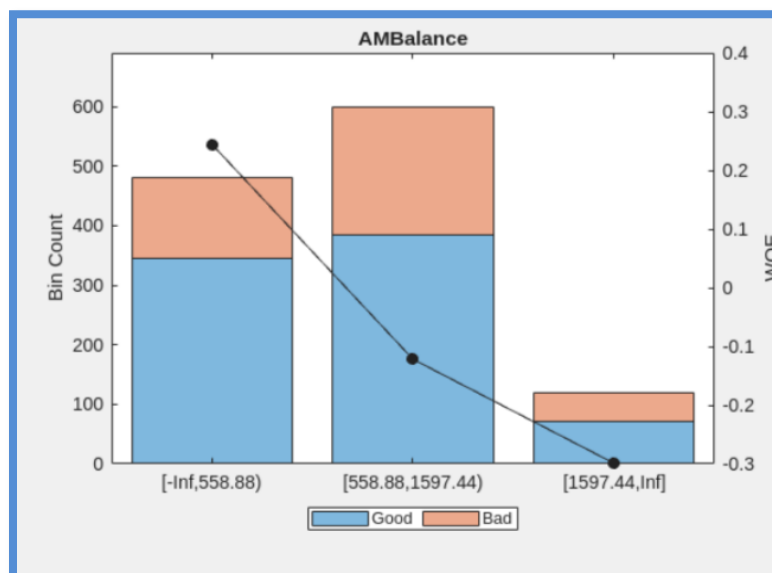
"Good" (tidak berisiko) dalam setiap bin. Sebagai contoh, pada bin $[-\text{Inf}, 558.88)$, ada 346 pelanggan yang dianggap "Good". Kolom "Bad" menunjukkan jumlah pelanggan yang dikategorikan sebagai "Bad" (berisiko) dalam setiap bin. Sebagai contoh, pada bin $[558.88, 1254.28)$, ada 171 pelanggan yang dianggap "Bad".

Kolom "Odds" menunjukkan rasio antara jumlah pelanggan "Good" dan "Bad" dalam setiap bin. Misalnya, pada bin $[-\text{Inf}, 558.88)$, odds adalah 2.5821, yang berarti rasio pelanggan "Good" terhadap "Bad" adalah sekitar 2.58:1. Kolom "WOE" (Weight of Evidence) mengukur pengaruh bin terhadap risiko. Nilai WOE positif menunjukkan lebih banyak pelanggan "Good", sementara nilai WOE negatif menunjukkan lebih banyak pelanggan "Bad". Sebagai contoh, pada bin $[-\text{Inf}, 558.88)$, nilai WOE adalah 0.2442, yang menunjukkan bahwa bin ini lebih cenderung memiliki lebih banyak pelanggan "Good" dibandingkan "Bad".

Kolom "InfoValue" menunjukkan seberapa efektif bin dalam memprediksi risiko. Semakin tinggi nilai InfoValue, semakin prediktif bin tersebut. Misalnya, pada bin $[-\text{Inf}, 558.88)$, InfoValue adalah 0.0228. Titik potong yang digunakan untuk membagi `AMBalance` ke dalam bin adalah 558.8, 1254.28, dan 1597.44, yang membentuk rentang bin yang berbeda.

```
cp(2) = []; % To merge bins 2 and 3
sc = modifybins(sc, 'AMBalance', 'CutPoints', cp);
plotbins(sc, 'AMBalance')
```

kode diatas menggabungkan beberapa bin pada `AMBalance` dan memvisualisasikan perubahan tersebut untuk memudahkan analisis. Berikut hasil visualisasinya :



Bin Count (Sumbu Y kiri) menggambarkan jumlah pelanggan pada setiap rentang nilai `AMBalance`. Bin pertama ($[-\text{Inf}, 558.88)$) memiliki jumlah pelanggan yang lebih banyak dibandingkan dengan bin kedua ($[558.88, 1597.44)$), yang mencatatkan jumlah pelanggan lebih sedikit. Sementara bin ketiga ($[1597.44, \text{Inf}]$) mencatatkan jumlah pelanggan yang paling sedikit. Bar biru menunjukkan pelanggan "Good" (tidak berisiko), sedangkan bar merah muda menunjukkan pelanggan "Bad" (berisiko). Pada bin pertama ($[-\text{Inf}, 558.88)$), jumlah pelanggan "Good" lebih banyak daripada "Bad". Di bin kedua ($[558.88, 1597.44)$), distribusi "Good" dan "Bad" lebih seimbang, meskipun ada sedikit lebih banyak pelanggan "Good". Bin ketiga ($[1597.44, \text{Inf}]$) menunjukkan proporsi "Bad" yang lebih tinggi, meskipun jumlah pelanggan secara keseluruhan lebih rendah.

Nilai WOE (Weight of Evidence, Sumbu Y kanan) yang ditunjukkan dengan titik hitam yang terhubung oleh garis, menggambarkan pengaruh setiap bin terhadap risiko. Pada bin pertama ($[-\text{Inf}, 558.88)$), nilai WOE positif menunjukkan lebih banyak pelanggan "Good". Pada bin kedua ($[558.88, 1597.44)$), nilai WOE menurun, menunjukkan pengurangan jumlah pelanggan "Good". Sedangkan pada bin ketiga ($[1597.44, \text{Inf}]$), nilai WOE negatif menunjukkan bahwa pelanggan pada rentang ini lebih banyak yang berisiko ("Bad"). Grafik ini menunjukkan hubungan antara nilai `AMBalance` dan risiko kredit, dengan WOE yang lebih tinggi menunjukkan lebih banyak pelanggan "Good" dan lebih rendah risiko kreditnya.

3. SESUAIKAN MODEL REGRESI LOGISTIK

Fungsi fitmodel digunakan untuk menyesuaikan model regresi logistik dengan data berbasis WOE. Di dalamnya, fitmodel secara otomatis mengelompokkan data pelatihan, mengonversinya ke nilai WOE, memetakan variabel respons sehingga nilai 'Good' diberi skor 1, dan menyusun model regresi logistik linier. Secara default, fitmodel menerapkan prosedur bertahap untuk menentukan prediktor mana yang sebaiknya dimasukkan dalam model.

```
sc = fitmodel(sc);
```

Kode `sc = fitmodel(sc);` pada MATLAB berarti variabel `sc` diproses menggunakan fungsi `fitmodel`, dan hasilnya disimpan kembali ke dalam variabel `sc`. Ini biasanya menunjukkan bahwa `sc` adalah data atau objek yang diberikan sebagai input untuk fungsi `fitmodel`, yang

kemudian menghasilkan model atau output yang telah diperbarui. Dalam konteks regresi logistik, `fitmodel(sc);` mungkin digunakan untuk menyesuaikan model regresi logistik dengan data `sc`, yang berisi variabel independen dan respons. Setelah pemrosesan, model yang sudah diperbarui disimpan kembali ke dalam `sc`, sehingga `sc` sekarang merepresentasikan model regresi logistik yang telah dilatih atau dioptimalkan berdasarkan data tersebut. Berikut hasil dari kode diatas :

```

1. Adding CustIncome, Deviance = 1490.8954, Chi2Stat = 32.545914, PValue = 1.1640961e-08
2. Adding TmWBank, Deviance = 1467.3249, Chi2Stat = 23.570535, PValue = 1.2041739e-06
3. Adding AMBalance, Deviance = 1455.858, Chi2Stat = 11.466846, PValue = 0.00070848829
4. Adding EmpStatus, Deviance = 1447.6148, Chi2Stat = 8.2432677, PValue = 0.0040903428
5. Adding CustAge, Deviance = 1442.06, Chi2Stat = 5.5547849, PValue = 0.018430237
6. Adding ResStatus, Deviance = 1437.9435, Chi2Stat = 4.1164321, PValue = 0.042468555
7. Adding OtherCC, Deviance = 1433.7372, Chi2Stat = 4.2063597, PValue = 0.040272676

Generalized linear regression model:
  logit(status) ~ 1 + CustAge + ResStatus + EmpStatus + CustIncome + TmWBank + OtherCC + AMBalance
Distribution = Binomial

Estimated Coefficients:
              Estimate      SE      tStat      pValue
-----
(Intercept)  0.7024      0.064    10.975  5.0407e-28
CustAge      0.61562     0.24783    2.4841  0.012988
ResStatus    1.3776     0.65266    2.1107  0.034799
EmpStatus    0.88592     0.29296    3.024   0.0024946
CustIncome   0.69836     0.21715    3.216   0.0013001
TmWBank      1.106       0.23266    4.7538  1.9958e-06
OtherCC      1.0933     0.52911    2.0662  0.038806
AMBalance    1.0437     0.32292    3.2322  0.0012285

1200 observations, 1192 error degrees of freedom
Dispersion: 1
Chi^2-statistic vs. constant model: 89.7, p-value = 1.42e-16

```

Gambar tersebut menunjukkan output dari fungsi `fitmodel` yang telah menyesuaikan model regresi logistik pada data yang diberikan. Berikut adalah penjelasan dari setiap bagian yang ditampilkan :

1. Di bagian atas, terlihat urutan penambahan variabel prediktor dalam model regresi logistik serta deviasi yang diperoleh setelah setiap variabel ditambahkan. Setiap kali variabel baru ditambahkan, ada nilai Chi2Stat dan P-value yang menunjukkan signifikansi dari penambahan variabel tersebut. Misalnya, ketika `CustIncome` ditambahkan, nilai Chi2Stat sebesar 32.54 dengan p-value sangat kecil (1.16e-08) menunjukkan bahwa penambahan variabel ini secara signifikan meningkatkan model.
2. Model Regresi Logistik merupakan model yang dihasilkan ditampilkan sebagai `logit(status) ~ ...`, yang menunjukkan bahwa respons (variabel dependen) `status`

dimodelkan menggunakan beberapa variabel independen (`CustAge`, `ResStatus`, `EmpStatus`, dll.) melalui fungsi logit, yang cocok untuk distribusi binomial/logistik.

3. Bagian Estimated Coefficients menampilkan estimasi koefisien untuk setiap variabel dalam model, serta Standard Error (SE), t-statistic (tStat), dan P-value. Setiap variabel memiliki kontribusi terhadap model yang direpresentasikan oleh nilai koefisiennya.

Misalnya :

- CustAge memiliki koefisien 0.61562 dengan p-value 0.012988, yang menunjukkan bahwa usia pelanggan (CustAge) berpengaruh signifikan dalam memprediksi `status`.
- EmpStatus memiliki koefisien 0.88592 dengan p-value yang sangat rendah (0.0002406), yang berarti status pekerjaan juga merupakan prediktor penting.
- P-value rendah (biasanya < 0.05) menunjukkan signifikansi dalam model. Variabel seperti `CustAge`, `ResStatus`, `EmpStatus`, `CustIncome`, `TmiBank`, dan `AMBalance` memiliki p-value signifikan, sehingga penting dalam model ini.

4. Statistik Model :

- Model ini memiliki 1200 observasi dan 1192 derajat kebebasan (error degrees of freedom).
- Chi²-statistic terhadap model konstan adalah 89.7 dengan p-value sangat kecil (1.42e-16), yang menunjukkan bahwa model ini secara signifikan lebih baik dibandingkan model tanpa variabel prediktor.

Secara keseluruhan, hasil ini menunjukkan bahwa variabel-variabel yang dipilih (seperti `CustAge`, `ResStatus`, `EmpStatus`, dll.) memiliki pengaruh yang signifikan dalam model regresi logistik untuk memprediksi `status`.

4. TINJAU DAN FORMAT POIN SCORECARD

Setelah model logistik disesuaikan, poin secara default tidak diskalakan dan diperoleh langsung dari kombinasi nilai WOE dan koefisien model. Fungsi `displaypoints` digunakan untuk merangkum poin pada kartu skor.

```
p1 = displaypoints(sc);  
disp(p1)
```

Kode ini memanggil fungsi untuk merangkum poin kartu skor dan kemudian menampilkan hasilnya untuk ditinjau. `p1 = displaypoints(sc);` : Baris ini menjalankan fungsi `displaypoints` pada objek atau data `sc`, yang dihasilkan dari model regresi logistik sebelumnya. Fungsi `displaypoints` akan menampilkan ringkasan poin kartu skor yang dihasilkan dari model yang disesuaikan. Hasilnya disimpan dalam variabel `p1`. `disp(p1);` : Baris ini menampilkan isi variabel `p1` di Command Window MATLAB. `disp` adalah fungsi MATLAB untuk menampilkan hasil atau output dari suatu variabel tanpa menambahkan informasi tambahan, sehingga output `p1` ditampilkan langsung dalam format yang mudah dibaca. Berikut hasil yang didapat dari kode tersebut :

Predictors	Bin	Points
{'CustAge' }	{'[-Inf,37)'	-0.15314
{'CustAge' }	{'[37,40)'	-0.062247
{'CustAge' }	{'[40,46)'	0.045763
{'CustAge' }	{'[46,58)'	0.22888
{'CustAge' }	{'[58,Inf]'	0.48354
{'CustAge' }	{'<missing>'	NaN
{'ResStatus' }	{'Tenant' }	-0.031302
{'ResStatus' }	{'Home Owner' }	0.12697
{'ResStatus' }	{'Other' }	0.37652
{'ResStatus' }	{'<missing>'	NaN
{'EmpStatus' }	{'Unknown' }	-0.076369
{'EmpStatus' }	{'Employed' }	0.31456
{'EmpStatus' }	{'<missing>'	NaN
{'CustIncome' }	{'[-Inf,29000)'	-0.45455
{'CustIncome' }	{'[29000,33000)'	-0.1037
{'CustIncome' }	{'[33000,42000)'	0.077768
{'CustIncome' }	{'[42000,47000)'	0.24406
{'CustIncome' }	{'[47000,Inf]'	0.43536
{'CustIncome' }	{'<missing>'	NaN
{'TmWBank' }	{'[-Inf,12)'	-0.18221
{'TmWBank' }	{'[12,45)'	-0.038279
{'TmWBank' }	{'[45,71)'	0.39569
{'TmWBank' }	{'[71,Inf]'	0.95074
{'TmWBank' }	{'<missing>'	NaN
{'OtherCC' }	{'No' }	-0.193
{'OtherCC' }	{'Yes' }	0.15868
{'OtherCC' }	{'<missing>'	NaN
{'AMBalance' }	{'[-Inf,558.88)'	0.3552
{'AMBalance' }	{'[558.88,1597.44)'	-0.026797
{'AMBalance' }	{'[1597.44,Inf]'	-0.21168
{'AMBalance' }	{'<missing>'	NaN

Tabel di atas menampilkan ringkasan poin scorecard untuk setiap prediktor dan kelompok data (Bin) yang dihasilkan dari model regresi logistik. Berikut adalah penjelasan tentang kolom-kolom yang ditampilkan:

- Predictors : Kolom ini menunjukkan variabel-variabel atau faktor prediktor dalam model, seperti CustAge, ResStatus, EmpStatus, CustIncome, TmWBank, OtherCC, dan

AMBalance. Setiap prediktor ini memiliki beberapa kategori atau rentang nilai (Bin) yang berbeda.

- Bin : Kolom ini mengelompokkan setiap prediktor ke dalam rentang atau kategori yang berbeda. Misalnya :
 - CustAge (usia pelanggan) dibagi menjadi rentang usia seperti "Up to 36," "37 to 39," "40 to 45," dan seterusnya.
 - ResStatus (status tempat tinggal) memiliki kategori seperti "Tenant", "Home Owner", dan "Other".
 - Kategori ``<missing>`` menunjukkan nilai-nilai yang tidak tersedia atau hilang dalam data.
- Points : Kolom ini menunjukkan poin kartu skor untuk setiap kelompok (Bin) dalam setiap prediktor. Poin-poin ini dihasilkan dari nilai WOE (Weight of Evidence) yang dikombinasikan dengan koefisien regresi. Poin positif menunjukkan pengaruh positif terhadap prediksi status, sedangkan poin negatif menunjukkan pengaruh negatif. Contohnya:
 - Pada CustAge, kelompok "Up to 36" memiliki poin -0.15314, yang menunjukkan pengaruh negatif terhadap status.
 - Pada EmpStatus, kategori "Employed" memiliki poin 0.31456, yang menunjukkan pengaruh positif.
 - Poin NaN pada `<missing>` berarti tidak ada poin yang dihitung untuk nilai yang hilang dalam kategori tersebut.

Secara keseluruhan, poin kartu skor ini membantu dalam interpretasi model. Setiap prediktor memberikan kontribusi poin yang berbeda tergantung pada kelompoknya, dan total dari semua poin ini dapat digunakan untuk memprediksi atau mengklasifikasikan `status` berdasarkan model yang sudah disesuaikan.

Untuk mengubah label bin, gunakan ``modifbins`` untuk mengubah label bin. Berikut kode nya :

```

sc = modifybins(sc,'CustAge','BinLabels',...
{'Up to 36' '37 to 39' '40 to 45' '46 to 57' '58 and up'});

sc = modifybins(sc,'CustIncome','BinLabels',...
{'Up to 28999' '29000 to 32999' '33000 to 41999' '42000 to 46999'
'47000 and up'});

sc = modifybins(sc,'TmWBank','BinLabels',...
{'Up to 11' '12 to 44' '45 to 70' '71 and up'});

sc = modifybins(sc,'AMBalance','BinLabels',...
{'Up to 558.87' '558.88 to 1597.43' '1597.44 and up'});

p1 = displaypoints(sc);
disp(p1)

```

Kode diatas digunakan untuk mengubah label bin dari beberapa variabel prediktor di dalam objek `sc` dengan menggunakan fungsi `modifybins`, kemudian menampilkan poin scorecard yang sudah diperbarui. Pertama, label bin untuk variabel `CustAge` diubah dalam objek `sc`, dengan label yang baru ditetapkan sebagai "Up to 36," "37 to 39," "40 to 45," "46 to 57," dan "58 and up." Selanjutnya, label bin `CustIncome` juga diperbarui dengan label baru, yaitu "Up to 28999," "29000 to 32999," "33000 to 41999," "42000 to 46999," dan "47000 and up."

Label bin untuk variabel `TmWBank` diubah menjadi "Up to 11," "12 to 44," "45 to 70," dan "71 and up." Demikian pula, label bin untuk variabel `AMBalance` diperbarui dengan rentang "Up to 558.87," "558.88 to 1597.43," dan "1597.44 and up."

Setelah semua perubahan label bin selesai, fungsi `displaypoints` digunakan untuk menampilkan ringkasan poin kartu skor dengan label bin yang baru, dan hasilnya ditampilkan di Command Window MATLAB melalui `disp(p1)`. Secara keseluruhan, kode ini memperbarui label bin pada beberapa variabel dalam model `sc` agar lebih mudah dibaca dan estetik, kemudian menampilkan ringkasan poin kartu skor yang telah diperbarui. Berikut hasil yang didapat :

Predictors	Bin	Points
{'CustAge' }	{'Up to 36' }	-0.15314
{'CustAge' }	{'37 to 39' }	-0.062247
{'CustAge' }	{'40 to 45' }	0.045763
{'CustAge' }	{'46 to 57' }	0.22888
{'CustAge' }	{'58 and up' }	0.48354
{'CustAge' }	{'<missing>' }	NaN
{'ResStatus' }	{'Tenant' }	-0.031302
{'ResStatus' }	{'Home Owner' }	0.12697
{'ResStatus' }	{'Other' }	0.37652
{'ResStatus' }	{'<missing>' }	NaN
{'EmpStatus' }	{'Unknown' }	-0.076369
{'EmpStatus' }	{'Employed' }	0.31456
{'EmpStatus' }	{'<missing>' }	NaN
{'CustIncome' }	{'Up to 28999' }	-0.45455
{'CustIncome' }	{'29000 to 32999' }	-0.1037
{'CustIncome' }	{'33000 to 41999' }	0.077768
{'CustIncome' }	{'42000 to 46999' }	0.24406
{'CustIncome' }	{'47000 and up' }	0.43536
{'CustIncome' }	{'<missing>' }	NaN
{'TmWBank' }	{'Up to 11' }	-0.18221
{'TmWBank' }	{'12 to 44' }	-0.038279
{'TmWBank' }	{'45 to 70' }	0.39569
{'TmWBank' }	{'71 and up' }	0.95074
{'TmWBank' }	{'<missing>' }	NaN
{'OtherCC' }	{'No' }	-0.193
{'OtherCC' }	{'Yes' }	0.15868
{'OtherCC' }	{'<missing>' }	NaN
{'AMBBalance' }	{'Up to 558.87' }	0.3552
{'AMBBalance' }	{'558.88 to 1597.43' }	-0.026797
{'AMBBalance' }	{'1597.44 and up' }	-0.21168
{'AMBBalance' }	{'<missing>' }	NaN

Perbedaan antara tabel ini dan tabel sebelumnya terletak pada label bin dari beberapa variabel prediktor, yang telah diubah menggunakan fungsi `modifybins` agar lebih deskriptif dan mudah dipahami. Berikut adalah perbedaan spesifik:

- **Label Bin yang Lebih Jelas :** Pada tabel sebelumnya, label bin mungkin menggunakan nama yang kurang informatif atau teknis. Setelah perubahan, label bin pada variabel seperti `CustAge`, `CustIncome`, `TmWBank`, dan `AMBBalance` menjadi lebih deskriptif, dengan rentang nilai yang jelas (misalnya, "Up to 36," "29000 to 32999," dll.). Contoh spesifiknya adalah `CustAge`, yang sekarang memiliki label bin seperti "Up to 36," "37 to 39," dan seterusnya, sedangkan sebelumnya label mungkin kurang spesifik atau tidak deskriptif.
- **Kemudahan Interpretasi :** Dengan label bin yang baru, tabel ini lebih mudah diinterpretasi oleh pengguna, karena setiap rentang atau kategori sekarang memiliki label yang menunjukkan nilai atau rentang sebenarnya. Hal ini membuat pembacaan dan pemahaman tabel lebih cepat dan intuitif.

- Tidak Ada Perubahan pada Kolom Points : Meskipun label bin diubah, nilai Points tetap sama seperti sebelumnya, karena nilai-nilai ini dihitung berdasarkan model regresi logistik yang sudah di-fit. Perubahan ini hanya mempengaruhi tampilan label bin, bukan hasil atau nilai poin dari model.

Secara keseluruhan, perubahan ini hanya meningkatkan estetika dan keterbacaan tabel, tanpa mempengaruhi hasil analisis atau poin kartu skor.

Poin biasanya diskalakan dan sering kali dibulatkan. Untuk melakukan ini, gunakan fungsi ``formatpoints``. Misalnya, Anda bisa menetapkan level poin target yang sesuai dengan target tingkat peluang serta menentukan poin yang dibutuhkan untuk menggandakan peluang (POINTS-TO-DOUBLE-THE-ODDS/PDO). Berikut perintah yang harus diberikan :

```
TargetPoints = 500;
TargetOdds = 2;
PDO = 50; % Points to double the odds

sc = formatpoints(sc, 'PointsOddsAndPDO', [TargetPoints TargetOdds
PDO]);
p2 = displaypoints(sc);
disp(p2)
```

Kode diatas digunakan untuk menskalakan dan membulatkan poin dalam model kartu skor dengan menetapkan target poin, peluang, dan poin yang dibutuhkan untuk menggandakan peluang (Points to Double the Odds atau PDO). Pertama, parameter target ditetapkan dengan nilai-nilai sebagai berikut:

- TargetPoints menetapkan target poin untuk model kartu skor sebesar 500, TargetOdds menentukan target peluang (odds) pada poin target yang diinginkan, yaitu 2 atau 2:1, dan `PDO` (Points to Double the Odds) menetapkan jumlah poin yang diperlukan untuk menggandakan peluang, yaitu 50 poin.
- Fungsi `formatpoints` kemudian digunakan untuk menskalakan poin dalam model kartu skor `sc` berdasarkan parameter `TargetPoints`, `TargetOdds`, dan `PDO`. Fungsi ini mengatur skala poin agar sesuai dengan target yang ditetapkan dan membulatkannya sesuai dengan skala yang diatur.

- Terakhir, fungsi `displaypoints` menampilkan poin kartu skor yang telah diskalakan dan dibulatkan, dan hasilnya ditampilkan di Command Window MATLAB menggunakan `\disp(p2)`. Secara keseluruhan, kode ini menyelaraskan poin dalam model kartu skor dengan parameter target yang telah ditentukan.

Predictors	Bin	Points
{'CustAge' }	{'Up to 36' }	53.239
{'CustAge' }	{'37 to 39' }	59.796
{'CustAge' }	{'40 to 45' }	67.587
{'CustAge' }	{'46 to 57' }	80.796
{'CustAge' }	{'58 and up' }	99.166
{'CustAge' }	{'<missing>' }	NaN
{'ResStatus' }	{'Tenant' }	62.028
{'ResStatus' }	{'Home Owner' }	73.445
{'ResStatus' }	{'Other' }	91.446
{'ResStatus' }	{'<missing>' }	NaN
{'EmpStatus' }	{'Unknown' }	58.777
{'EmpStatus' }	{'Employed' }	86.976
{'EmpStatus' }	{'<missing>' }	NaN
{'CustIncome' }	{'Up to 28999' }	31.497
{'CustIncome' }	{'29000 to 32999' }	56.805
{'CustIncome' }	{'33000 to 41999' }	69.896
{'CustIncome' }	{'42000 to 46999' }	81.891
{'CustIncome' }	{'47000 and up' }	95.69
{'CustIncome' }	{'<missing>' }	NaN
{'TmWBank' }	{'Up to 11' }	51.142
{'TmWBank' }	{'12 to 44' }	61.524
{'TmWBank' }	{'45 to 70' }	92.829
{'TmWBank' }	{'71 and up' }	132.87
{'TmWBank' }	{'<missing>' }	NaN
{'OtherCC' }	{'No' }	50.364
{'OtherCC' }	{'Yes' }	75.732
{'OtherCC' }	{'<missing>' }	NaN
{'AMBBalance' }	{'Up to 558.87' }	89.908
{'AMBBalance' }	{'558.88 to 1597.43' }	62.353
{'AMBBalance' }	{'1597.44 and up' }	49.016
{'AMBBalance' }	{'<missing>' }	NaN

Tabel ini menunjukkan hasil poin scorecard setelah proses penskalaan dan pembulatan menggunakan parameter yang telah ditetapkan, yaitu `\TargetPoints`, `\TargetOdds`, dan `\PDO`. Dibandingkan dengan dua tabel sebelumnya, berikut perbedaan yang utama :

- Skala Poin yang Ditingkatkan : Pada tabel sebelumnya, poin scorecard memiliki nilai yang lebih kecil (misalnya, di sekitar -0.15 hingga 0.95), sementara di tabel ini, poin telah ditingkatkan skalanya. Nilai poin sekarang berada di kisaran puluhan hingga ratusan (contoh: "53.239" untuk `\CustAge` di kategori "Up to 36"). Penskalaan ini dilakukan untuk menyesuaikan poin scorecard dengan `\TargetPoints` dan `\TargetOdds` yang ditetapkan, membuat poin lebih mudah diinterpretasi dalam skala yang lebih besar.

- Pembulatan Poin : Poin di tabel ini telah dibulatkan sesuai dengan skala yang diatur menggunakan `PDO`, sehingga nilai poin lebih konsisten dan tidak memiliki desimal kecil seperti pada tabel awal. Ini membantu dalam penyajian yang lebih rapi dan lebih mudah dibaca, terutama jika kartu skor digunakan untuk evaluasi manual.
- Interpretasi dan Konsistensi : Peningkatan skala dan pembulatan ini membuat poin scorecard lebih konsisten dan memberikan interpretasi yang lebih intuitif dalam menentukan tingkat risiko atau peluang berdasarkan model. Poin yang lebih tinggi menunjukkan pengaruh yang lebih kuat terhadap prediksi `status`, sedangkan poin lebih rendah menunjukkan pengaruh yang lebih lemah, dalam skala yang lebih mudah dipahami oleh pengguna akhir.

Secara keseluruhan, tabel ini merupakan versi poin scorecard yang telah disesuaikan dan dibulatkan, sehingga lebih user-friendly dan siap untuk digunakan dalam aplikasi atau interpretasi langsung dari model.

5. SCORE DATA

Fungsi `score` menghitung skor untuk data pelatihan. Data tambahan, seperti data validasi, juga dapat diberikan sebagai input opsional untuk dihitung skornya. Poin per prediktor untuk setiap pelanggan disediakan sebagai output opsional.

```
[Scores,Points] = score(sc);
disp(Scores(1:10))
```

Kode ini menghitung skor untuk data menggunakan fungsi `score` dan menampilkan skor untuk 10 data pertama. Pertama, fungsi `score` dijalankan pada objek `sc` (model kartu skor), menghasilkan dua output: `Scores`, yang merupakan total skor untuk setiap data dalam data pelatihan, dan `Points`, yang berisi poin per prediktor untuk setiap data sebagai output opsional. Kemudian, fungsi `disp` digunakan untuk menampilkan 10 skor pertama dari variabel `Scores` di Command Window MATLAB. Secara keseluruhan, kode ini menghitung skor berdasarkan model kartu skor `sc` dan menampilkan hasilnya untuk 10 data pertama. Berikut hasil yang didapat dari kode tersebut :

```
528.2044
554.8861
505.2406
564.0717
554.8861
586.1904
441.8755
515.8125
524.4553
508.3169
```

Hasil yang ditampilkan menunjukkan 10 skor pertama yang dihitung oleh fungsi `score` pada data. Setiap nilai skor mencerminkan total poin yang diperoleh oleh masing-masing data pelanggan berdasarkan model kartu skor yang diterapkan. Sebagai contoh, skor seperti 528.2044, 554.8861 dan 505.2406 menunjukkan bahwa setiap data memiliki skor total yang berbeda. Skor-skor ini dihasilkan dari penjumlahan poin yang diberikan oleh setiap prediktor dalam model, dan dapat digunakan untuk mengevaluasi atau mengklasifikasikan data sesuai dengan target yang ditentukan dalam model.

```
disp(Points(1:10,:))
```

Kode ini berfungsi untuk menampilkan poin per prediktor bagi 10 data pertama. Fungsi `disp` digunakan untuk menampilkan nilai dari variabel `Points` pada baris 1 hingga 10, yang berisi poin per prediktor untuk setiap pelanggan dalam 10 data pertama. Notasi `Points(1:10,:)` menunjukkan bahwa kode ini mengambil 10 baris pertama dari `Points` pada semua kolom. Hasil yang ditampilkan akan memperlihatkan kontribusi masing-masing prediktor terhadap total skor untuk setiap data pada 10 data pertama, memberikan rincian poin per variabel atau prediktor dalam model. Berikut hasil yang ditampilkan :

CustAge	ResStatus	EmpStatus	CustIncome	TmWBank	OtherCC	AMBance
80.796	62.028	58.777	95.69	92.829	75.732	62.353
99.166	73.445	86.976	95.69	61.524	75.732	62.353
80.796	62.028	86.976	69.896	92.829	50.364	62.353
80.796	73.445	86.976	95.69	61.524	75.732	89.908
99.166	73.445	86.976	95.69	61.524	75.732	62.353
99.166	73.445	86.976	95.69	92.829	75.732	62.353
53.239	73.445	58.777	56.805	61.524	75.732	62.353
80.796	91.446	86.976	95.69	61.524	50.364	49.016
80.796	62.028	58.777	95.69	61.524	75.732	89.908
80.796	73.445	58.777	95.69	61.524	75.732	62.353

Hasil yang ditampilkan menunjukkan kontribusi poin dari setiap prediktor untuk masing-masing pelanggan pada 10 data pertama. Tabel ini merinci skor total menjadi poin per prediktor, memberikan gambaran tentang bagaimana setiap variabel berkontribusi terhadap skor akhir. Penjelasan dari setiap kolom dalam tabel ini adalah sebagai berikut:

- CustAge : Kolom ini menampilkan poin yang diberikan oleh variabel `CustAge` (usia pelanggan) untuk tiap data, dengan nilai yang bervariasi sesuai dengan kategori usia pelanggan.
- ResStatus : Kolom ini berisi poin berdasarkan `ResStatus` (status tempat tinggal) pelanggan, seperti tenant atau pemilik rumah, yang memengaruhi skor total.
- EmpStatus : Kolom ini menunjukkan poin dari `EmpStatus` (status pekerjaan), yang berkontribusi sesuai dengan kategori pekerjaan pelanggan, misalnya apakah pelanggan bekerja atau tidak.
- CustIncome : Kolom ini memperlihatkan poin berdasarkan `CustIncome` (pendapatan pelanggan), di mana rentang pendapatan yang berbeda memberikan nilai kontribusi yang berbeda pula.
- TmWBank : Kolom ini menampilkan poin dari `TmWBank` (lama menjadi nasabah bank), di mana nasabah dengan masa keanggotaan lebih lama cenderung memiliki skor yang lebih tinggi.
- OtherCC : Kolom ini menunjukkan poin kontribusi dari `OtherCC`, yang menunjukkan apakah pelanggan memiliki kartu kredit lain.
- AMBalance : Kolom ini berisi poin berdasarkan `AMBalance` (saldo rata-rata), dengan kontribusi yang sesuai dengan jumlah saldo rata-rata pelanggan.

Setiap baris di tabel ini mewakili satu data pelanggan, dengan nilai di setiap kolom menunjukkan poin yang diberikan oleh masing-masing prediktor. Skor total setiap data merupakan hasil penjumlahan nilai di semua kolom. Tabel ini memudahkan pemahaman mengenai seberapa besar setiap faktor atau prediktor memengaruhi skor akhir untuk setiap pelanggan.

6. HITUNG PROBABILITAS GAGAL BAYAR

Gunakan fungsi `probdefault` untuk menghitung kemungkinan gagal bayar. Berikut kode nya :

```
pd = probdefault(sc);
```

Kode ini menggunakan fungsi `probdefault` untuk menghitung probabilitas gagal bayar berdasarkan model yang disesuaikan dalam objek `sc`. Hasil dari perhitungan ini disimpan dalam variabel `pd`, yang berisi nilai probabilitas gagal bayar untuk setiap data yang dianalisis.

Untuk menentukan probabilitas menjadi "Good" dan membuat plot yang menunjukkan hubungan antara peluang yang diprediksi dan skor yang telah diformat. Kemudian menganalisis visual untuk memastikan bahwa poin target dan peluang target sesuai, serta bahwa hubungan poin dengan the points to double the odds (PDO) berlaku. Berikut kode yang dibutuhkan untuk memvisualisasikannya :

```
ProbGood = 1-pd;
PredictedOdds = ProbGood./pd;

figure
scatter(Scores,PredictedOdds)
title('Predicted Odds vs. Score')
xlabel('Score')
ylabel('Predicted Odds')

hold on

xLimits = xlim;
yLimits = ylim;

% Target points and odds
plot([TargetPoints TargetPoints],[yLimits(1) TargetOdds],'k:')
plot([xLimits(1) TargetPoints],[TargetOdds TargetOdds],'k:')

% Target points plus PDO
plot([TargetPoints+PDO TargetPoints+PDO],[yLimits(1)
2*TargetOdds],'k:')
plot([xLimits(1) TargetPoints+PDO],[2*TargetOdds 2*TargetOdds],'k:')

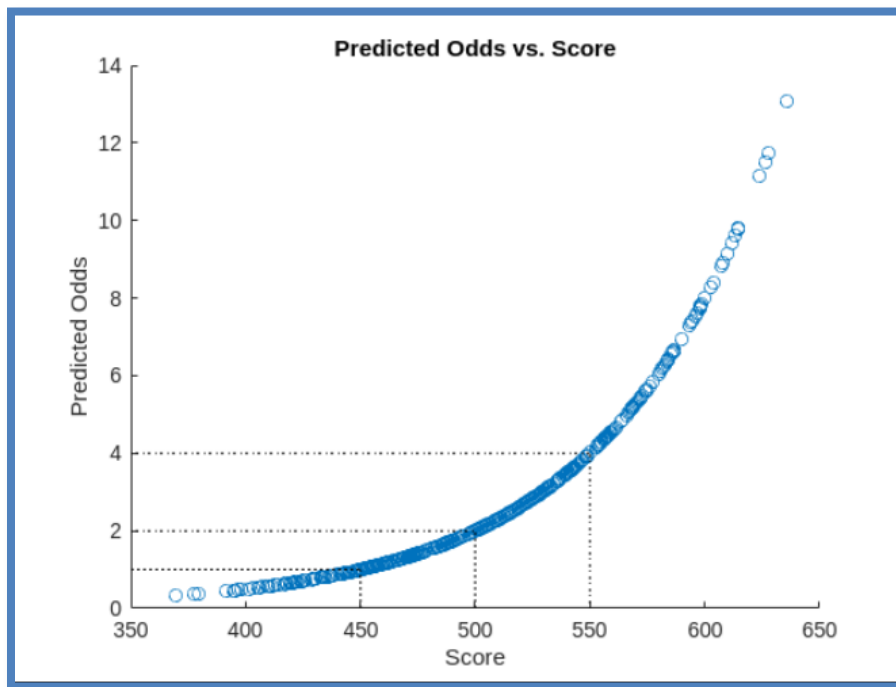
% Target points minus PDO
plot([TargetPoints-PDO TargetPoints-PDO],[yLimits(1)
TargetOdds/2],'k:')
plot([xLimits(1) TargetPoints-PDO],[TargetOdds/2 TargetOdds/2],'k:')

hold off
```

Kode ini digunakan untuk memvisualisasikan hubungan antara skor yang diprediksi dan odds yang diprediksi, serta menunjukkan target poin dan hubungan penggandaan peluang (PDO) pada grafik. Pertama, probabilitas untuk "Baik" dihitung dengan mengurangi nilai `pd` dari 1. Kemudian, odds prediksi dihitung dengan membagi probabilitas "Baik" dengan probabilitas gagal bayar (`pd`). Setelah itu, plot sebaran dibuat antara skor pelanggan (`Scores`) dan odds yang diprediksi (`PredictedOdds`) menggunakan fungsi `scatter`. Grafik ini diberi label dan judul yang sesuai untuk menggambarkan hubungan antara skor dan odds prediksi.

Garis kemudian ditambahkan pada plot untuk menunjukkan `TargetPoints` dan `TargetOdds`, yaitu target nilai poin dan odds. Garis-garis ini menunjukkan di mana target poin dan odds berada dalam grafik. Selanjutnya, garis ditambahkan untuk menunjukkan `TargetPoints + PDO` dan `TargetPoints - PDO`, yang menggambarkan efek dari menggandakan peluang (PDO) terhadap target poin. Garis-garis ini juga digambarkan dengan garis putus-putus untuk memberikan gambaran visual tentang pengaruh PDO pada odds.

Akhirnya, `hold off` digunakan untuk mengakhiri penambahan elemen ke dalam grafik, menyelesaikan visualisasi. Grafik ini memberikan gambaran yang jelas tentang bagaimana skor dan odds prediksi saling terkait, serta memastikan bahwa hubungan antara poin target dan odds target, bersama dengan pengaruh PDO, terlihat dengan jelas.



Hasil grafik ini menunjukkan hubungan antara Predicted Odds (peluang yang diprediksi) dan Score (skor). Grafik ini memperlihatkan sebaran data yang menggambarkan bagaimana odds berubah seiring dengan meningkatnya skor. Berikut penjelasan dari elemen-elemen yang ada pada grafik :

- Sumbu X (Score) : Menunjukkan skor yang diberikan untuk masing-masing data, dengan rentang antara 350 hingga 650. Skor ini dihasilkan dari model kartu skor yang dihitung sebelumnya.
- Sumbu Y (Predicted Odds) : Menunjukkan peluang yang diprediksi, dengan nilai yang meningkat seiring dengan skor. Odds ini menggambarkan rasio peluang untuk gagal bayar terhadap peluang untuk "Good".
- Titik-titik Biru : Setiap titik mewakili satu data pelanggan, dengan posisi pada grafik menunjukkan skor dan odds yang diprediksi untuk data tersebut.
- Garis Dotted (Putus-putus) : Garis ini menunjukkan level tertentu pada grafik yang terkait dengan target poin dan odds. Garis putus-putus ini membantu menunjukkan di mana target-target tersebut terletak, memudahkan analisis visual hubungan antara skor dan odds yang diprediksi, serta memverifikasi apakah target yang ditentukan sebelumnya tercapai.

Secara keseluruhan, grafik ini menggambarkan bagaimana odds yang diprediksi meningkat seiring dengan meningkatnya skor, dengan penambahan garis untuk menunjukkan target poin dan odds yang relevan, serta hubungan antara poin dan penggandaan peluang (PDO).