« بسم الله الرحمن الرحیم »

درس : داده کاوی

استاد : دکتر فقیهی

گردآورنده: آرش صادقی بابلان

شماره دانشجویی: ۴۰۰۴۲۲۱۱۶

گزارش تحلیل داده‌های هزینه و درآمد خانوار ایران، شهرستان تهران

مقدمه

پژوهش حاضر با هدف بررسی رابطه بین متغیرهای جمعیت‌شناختی، سبک زندگی و وضعیت ثروت خانواده، که توسط متغیر دودویی «DAHAK10» نمایش داده می‌شود، انجام شده است. مجموعه داده‌های مورد استفاده در این پروژه شامل طیف گسترده‌ای از متغیرها از جمله جنسیت، سطح تحصیلات و وضعیت شغلی سرپرست خانوار، همچنین نوع تصرف محل اسکان، هزینه‌های خانوار و وجود برخی از اقلام خانگی است که تصویری جامع از وضعیت هر فرد ارائه می‌کند.

هدف از این تحلیل، روشن کردن متغیرها و شرایطی است که با ثروت و رفاه مالی مرتبط است. این اطلاعات می‌تواند برای سیاست‌گذاران و محققان ارزشمند باشد، زیرا به گسترش دانش در مورد اقتصاد و رفاه مالی کمک می‌کند و ممکن است الهام‌بخش تحقیقات و تحلیل‌های بیشتر در این زمینه باشد.

روش‌ها

برای دستیابی به هدف این مطالعه، یک فرآیند جامع تحلیل اکتشافی داده‌ها شامل استفاده از مصورسازی‌های مختلف و روش‌های آماری انجام شده است. این روش‌ها برای به دست آوردن درک عمیق‌تر از داده‌ها، شناسایی الگوها و روابط بین متغیرها و کشف بینش در مورد عوامل مرتبط با ثروت و رفاه مالی استفاده شد.

علاوه بر فرآیند تحلیل اکتشافی داده‌ها، مدل‌های آماری و یادگیری ماشینی برای پیشگویی و به دست آوردن درک دقیق‌تر از روابط بین متغیرها به کار گرفته شد. مدل های مورد استفاده در این مطالعه شامل درخت تصمیم، رگرسیون لجستیک و شبکه‌های عصبی بود. هر مدل به دلیل توانایی‌اش در پرداختن به جنبه‌های مختلف داده‌ها و ارائه بینش‌های مکمل انتخاب شد.

از مدل درخت تصمیم برای کشف روابط بین متغیرهای مستقل و متغیر وابسته "DAHAK10" استفاده شد. این مدل به ویژه برای شناسایی مهم‌ترین متغیرها در توضیح تغییرات متغیر وابسته مفید است.

از مدل رگرسیون لجستیک برای پیش‌بینی احتمال تعلق افراد به کلاس «DAHAK10» بر اساس ویژگی‌های جمعیت‌شناختی و سبک زندگی آن‌ها استفاده شد. رگرسیون لجستیک یک روش آماری پرکاربرد است که به ویژه برای پیش‌بینی نتایج دودویی مانند "DAHAK10" مناسب است.

مدل شبکه عصبی برای ساخت یک مدل پیش‌بینی که روابط پیچیده بین متغیرهای مستقل و متغیر وابسته را در نظر می‌گیرد، استفاده شد. این مدل به ویژه برای کشف روابط غیرخطی بین متغیرها مفید است که با استفاده از روش‌های دیگر به راحتی قابل تشخیص نیستند.

هر یک از این مدل‌ها برای کشف بینش‌های منحصربه‌فرد در مورد روابط بین متغیرها استفاده شد و نتایج مقایسه و ارزیابی شدند تا مشخص شود کدام مدل‌ها بهترین پیش‌بینی‌ها را ارائه می‌کنند. یافته‌های حاصل از این مدل‌ها در بخش‌های بعدی به تفصیل ارائه و مورد بحث قرار خواهد گرفت.

علاوه بر مدل‌های یادگیری ماشین، نمودارها و مصورسازی‌های مختلفی برای به دست آوردن درک عمیق‌تر از داده‌ها و شناسایی الگوها و روابط بین متغیرها استفاده شد. این مصورسازی ها شامل رسم بافت نگاشت، نمودار پراکنش و غیره هستند. استفاده از این تجسم‌ها درک جامع‌تری از داده‌ها را فراهم و به کشف بینش‌هایی کمک کرد که ممکن بود تنها با استفاده از روش‌های آماری از قلم افتاده باشند.

یافته‌ها:

نتایج این مطالعه در بخش‌های بعدی به تفصیل ارائه و مورد بحث قرار خواهد گرفت. یافته‌ها شامل بینش‌ها و الگوهای کلیدی خواهد بود که از طریق فرآیندهای کاوش و مدل‌سازی داده‌ها و همچنین پیامدهای این نتایج کشف شده‌اند.

نتیجه:

در نتیجه، این مطالعه با بررسی رابطه بین متغیرهای جمعیت‌شناختی و سبک زندگی و وضعیت ثروت یک فرد، به کسب دانش بیشتر در مورد ثروت و رفاه مالی کمک می‌کند. نتایج این تحلیل می‌تواند به تحقیقات آینده در این زمینه کمک کند و پیامدهایی برای سیاست‌گذاران و افراد به طور یکسان داشته باشد.

فصل ۱: مقدمه‌ای بر داده‌ها و پاکسازی آن‌ها

۱-۱ مقدمه

اولین قدم در انجام یک پروژه داده‌کاوی موفق این است که اطمینان حاصل شود که داده‌ها تمیز و به خوبی آماده شده‌اند. در این فصل، ما یک نمای کلی از فرآیند پاکسازی داده‌ها که بر روی مجموعه داده خام انجام شده است ارائه می‌کنیم تا مطمئن شویم که برای تجزیه و تحلیل آماده است. تمرکز این فصل ارائه درک روشنی از مشکلات موجود در مجموعه داده خام و نحوه رسیدگی به آنها از طریق فرآیند پاکسازی داده است.

در ابتدا، مجموعه داده خام از یک منبع خارجی دریافت شد که دارای مشکلات مختلفی از جمله تعداد کمی نام‌گذاری غلط و مقادیر گم‌شده بود. برای اینکه مجموعه داده قابل استفاده باشد، لازم بود داده‌ها تمیز و آماده شوند تا بتوان آن‌ها را تجزیه و تحلیل کرد.

پس از تکمیل فرآیند پاکسازی داده‌ها، مرحله بعدی ایجاد ستون DAHAK10 بود. ستون DAHAK10 یک متغیر دودویی است که نشان می‌دهد یک خانواده از نظر درآمد جزو دهک دهم است یا خیر. این ستون با استفاده از متغیرهای مختلف در مجموعه داده ایجاد شده و به عنوان متغیر وابسته در تحلیل، مورد استفاده قرار گرفته است.

در ادامه، یک جدول جامع ایجاد شد که شامل تمام متغیرهای مجموعه داده است و توضیح مختصری در مورد هر متغیر ارائه می‌دهد. جدول شامل اطلاعات زیر برای هر متغیر است

نام متغیر (variable name):

با ارائه یک نمای کلی از فرآیند پاکسازی داده‌ها و متغیرهای موجود در مجموعه داده، این فصل یک پایه محکم برای فصل‌های بعدی که در آن تجزیه و تحلیل داده‌ها انجام خواهد شد، فراهم می‌کند.

۱-۲ مشکلات داده‌ها

در مراحل اولیه بررسی داده‌ها، اختلافی بین توضیحات ستون‌های ارائه شده و داده‌های موجود مشخص شد. با بررسی بیشتر مشخص شد که ستون DARAMAD.M.KH.5 نیاز به تغییر نام به DARAMAD.M.KH.4 دارد. به منظور درک بهتر الگوهای هزینه و درآمد افراد، تعیین ماهانه و یا سالانه بودن متغیرهای مختلف ضروری بود. برای تسهیل این امر، ستون جدیدی با جمع تمام منابع درآمدی ایجاد شد که به ما امکان داد دهک‌ها را محاسبه و در نهایت ستون جدیدی به نام «DAHAK» ایجاد کنیم.

با ستون 'DAHAK'، یک متغیر دودویی 'DAHAK10' برای نشان دادن وضعیت ثروت افراد در جمعیت ایجاد شد. در ستون 'DAHAK10' در صورتی که مقدار 'DAHAK' یک فرد۱۰ باشد، مقدار ۱ و در غیر این صورت مقدار ۰ اختصاص داده می‌شود. این ستون جدید متغیر رسته‌ای اصلی است که ما باید با استفاده از دیگر متغیرها پیش‌بینی کنیم.

۱-۲ معرفی متغیرها

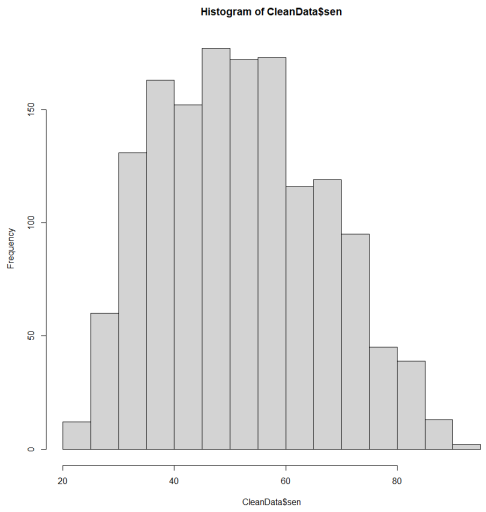
تعریف متغیر	نام متغیر
کد خانوار	C.KH
کد استان	C.Sh
جنس سرپرست خانوار	JENS
سن سرپرست خانوار	SEN
سرپرست خانوار سواد دارد یا ندارد؟	SAVAD
مدرک تحصیلی سرپرست خانوار	M.T.S
وضعیت فعالیت سرپرست خانوار	V.F.S
نحوه تصرف منزل مسکونی	N.T.M
تعداد اتاق در اختیار	T.O
سطح زیر بنای محل سکونت	S.Z
مصالح عمده بنای محل سکونت	M.O.B
اتومبیل شخصی	OTO
موتورسیکلت	MO
دوچرخه	DO
ضبط صوت	ZABT
تلویزیون رنگی	TV
انواع یارانه و تبلت	PC
تلفن همراه	TEL.H
اجاق گاز	OJAGH.GAZ
جارو برقی	JAROO.B
ماشین لباسشویی	M.LEBAS
چرخ خیاطی	CHARKH.KH
پنکه	PANKE
ماشین ظرفشویی	M.ZARF
هزینه‌های خوراکی و نوشیدنی خانوار در یک ماه گذشته	H.KHORAKI.NOOSHIDANI
هزینه ارتباطات خانوار در یک ماه گذشته	H.ERTEBATAT
هزینه‌های بهداشتی خانوار در یک ماه گذشته	H.BEHDASHT
هزینه‌های حمل و نقل خانوار در یک ماه گذشته	H.HAMLONAGHL
هزینه کالاها یا خدمات متفرقه خانواردر یک ماه گذشته	H.KALA.MOT
هزینه‌های مسکن- آب، سوخت، روشنایی و...	H.MASKAN
هزینه های مبلمان و لوازم خانگی و نگهداری‌های معمول آنها در ماه گذشته	H.MOBLEMAN
هزینه‌های پوشاک خانوار در یک ماه گذشته	H.POOSHAK
هزینه خرید کالای بادوام خانوار در ۱۲ ماه گذشته	KHARID.KALA.BADAVAM

H.SARMAYEGOZARI	هزینه سرمایه‌گذاری خانوار در ۱۲ ماه گذشته
M.DARAMAD.NAKH	مجموع درآمدهای ناخالص مستمر و غیر مستمر ۱۲ ماه گذشته اعضای شاغل خانوار قبل از کسورات
HOOGHOOGH.MOSTAMAR	مزد و حقوق مستمر ۱۲ ماه گذشته
GH.MOSTAMAR	مزایای غیر مستمر ۱۲ ماه گذشته
DARYAFTI.NAKH.F	دریافتی ناخالص از فروش
DARAMAD.M.KH.1	درآمد حاصل از اجاره محل کسب، باغ، زمین، مستغلات منزل، حق کسب و کار، اموال منقول و غیرمنقول و نظایر آن
DARAMAD.M.KH.2	درآمد حاصل از پس‌انداز سپرده ثابت، سهام، بیمه و نظایر آن
DARAMAD.M.KH.3	حقوق بازنشستگی در ۱۲ ماه گذشته
DARAMAD.M.KH.4	کمک هزینه تحصیلی در ۱۲ ماه گذشته

فصل ۲: تصویرسازی داده‌ها

در این فصل با رسم برخی نمودارها و به دست آوردن شاخص‌های آماری تلاش می‌کنیم تا درک بهتری از داده‌ها به‌دست آوریم.

۲-۱ بافت نگار سن

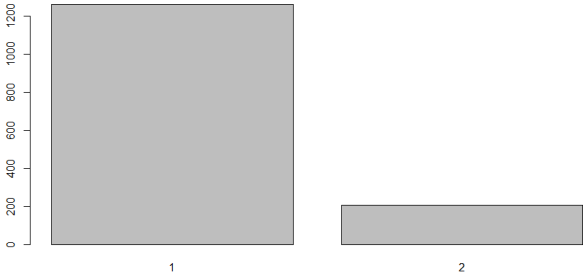


این هیستوگرام بینشی از توزیع سنی سرپرستان خانوار ارائه می دهد. داده‌ها تقریباً از توزیع نرمال پیروی می‌کنند، البته با کمی چولگی به راست، که نشان می‌دهد اکثریت سرپرستان خانوارها در سنین میانسالی خود هستند. مقدار قابل توجهی از سنین بین ۳۰ تا ۶۰ سال را می توان مشاهده کرد که بزرگترین جمعیت سرپرستان خانوار را تشکیل می دهد.

این اطلاعات از چند جهت مفید است. اولاً، درک کلی از جمعیت شناسی سرپرستان خانوار ارائه می دهد و می تواند سیاست ها و برنامه هایی را با هدف خدمت به این جمعیت خاص ارائه دهد. ثانیاً، می توان از آن برای شناسایی روندها و الگوهای بالقوه در داده ها، مانند افزایش یا کاهش تعداد سرپرستان خانوار در گروه های سنی خاص در طول زمان استفاده کرد.

در خاتمه، هیستوگرام توزیع سنی سرپرستان خانواده ابزار ارزشمندی برای درک بهتر ترکیب جمعیتی این جمعیت مهم است.

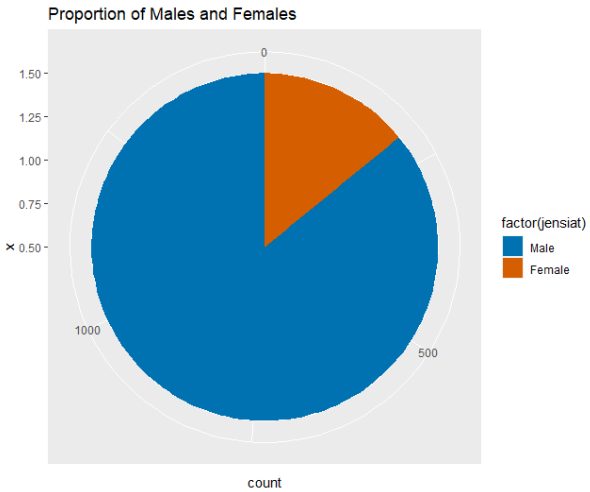
۲-۲ نمودار میله‌ای سواد



این نمودار میله ای مقایسه ای از وضعیت تحصیلی سرپرستان خانوارهای شهر تهران را ارائه می دهد. داده ها نشان می دهد که اکثریت سرپرستان خانوار، بیش از ۱۲۰۰ نفر، به نوعی آموزش دیده اند. از سوی دیگر حدود۲۰۰ نفر از سرپرستان خانوار آموزش تحصیلی ندیده اند و به مدرسه نرفته اند.

این اطلاعات در شناخت وضعیت تحصیلی سرپرستان خانوار در شهر تهران بسیار مهم است که می تواند سیاست ها و برنامه‌هایی را با هدف ارتقا و بهبود آموزش در این جمعیت ارائه دهد. نسبت بالای سرپرست خانوارهای تحصیل کرده نشانگر مثبتی از سطح تحصیلات کلی جمعیت است، در حالی که نسبت پایین افراد تحصیلکرده نیاز به تلاش های هدفمند برای افزایش فرصت ها و دسترسی های آموزشی را برجسته می کند.

۲-۳ نمودار دایره‌ای جنسیت

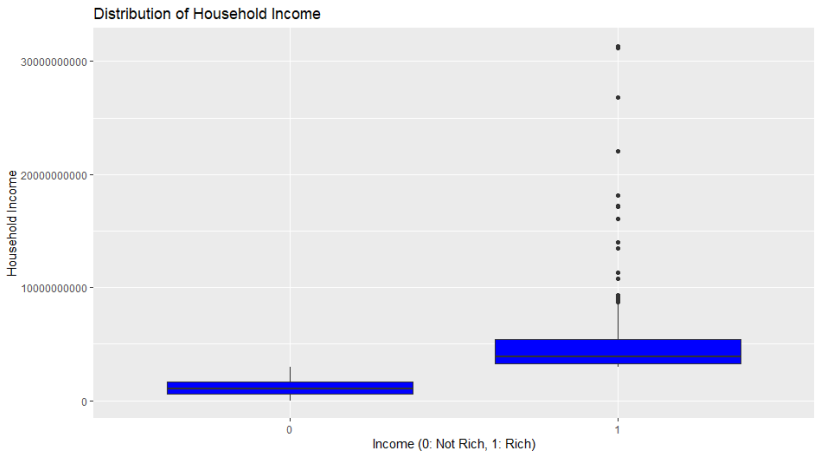


این نمودار دایره ای نشان دهنده توزیع جنسیتی سرپرستان خانوار است. داده ها نشان می دهد که ۸۸ درصد از سرپرستان خانوارها مرد و تنها ۱۲ درصد زن هستند.

این اطلاعات بسیار مهم است زیرا نشان می‌دهد که زنان در پست‌های رهبری در خانواده‌ها به صورت کم‌رنگ حضور دارند، که می‌تواند نیاز سیاست‌هایی را با هدف ارتقای برابری جنسیتی و توانمندسازی زنان بیان کند. حضور کم زنان به عنوان سرپرست خانوار، نیاز به تلاش های هدفمند برای افزایش نمایندگی زنان و رسیدگی به نابرابری های مبتنی بر جنسیت را برجسته می کند.

در پایان، نمودار دایره ای تصویر روشنی از توزیع جنسیتی سرپرستان خانوار ارائه می دهد و اهمیت ترویج برابری جنسیتی و توانمندسازی زنان در این جمعیت را برجسته می کند.

۴-۲ نمودار جعبه‌ای برای مقایسه درآمد دهک دهم با سایر دهک‌ها



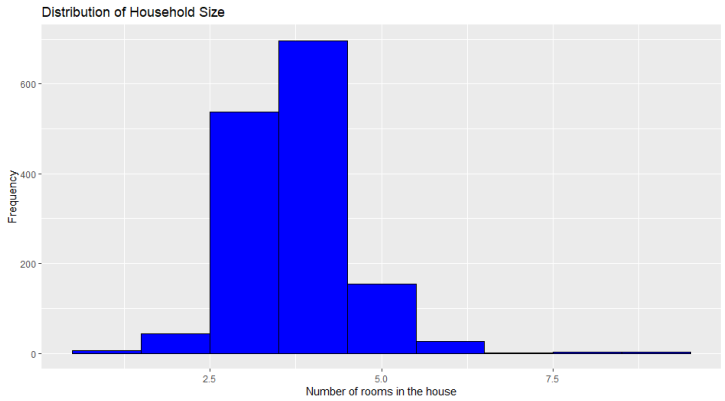
این نمودار جعبه ای درآمدهای خانواده ها را بین دهک دهم (ثروتمندترین خانواده ها) و باقی (دهک اول تا نهم) مقایسه می کند. بدیهی است که ثروتمندترین خانواده ها به طور قابل توجهی درآمد بالاتری نسبت به بقیه دارند.

با این حال، طرح یک مشاهدات جالب را نیز برجسته می کند: به نظر می رسد اختلاف درآمد در میان ثروتمندترین خانواده‌ها در مقایسه با بقیه جمعیت بسیار بیشتر است. این نشان می دهد که در حالی که میانگین درآمد در میان دهک های ۱۰ بالا به طور قابل توجهی بیشتر است، اما در میان این گروه، توزیع درآمد واریانس بیشتری دارد.

این اطلاعات در درک توزیع درآمد و نابرابری در بین جمعیت بسیار مهم است. واریانس بالاتر درآمد در میان ثروتمندترین خانواده‌ها، نیاز به تلاش‌های هدفمند برای رسیدگی به نابرابری درآمد و اطمینان از توزیع یکنواخت‌تر مزایای رشد اقتصادی را برجسته می‌کند.

در نتیجه، نمودار جعبه مقایسه جامعی از درآمد خانواده بین ثروتمندترین و بقیه جمعیت ارائه می دهد و اهمیت پرداختن به نابرابری درآمد را برجسته می کند.

۲-۵ بافت‌نگار تعداد اعضای خانوار

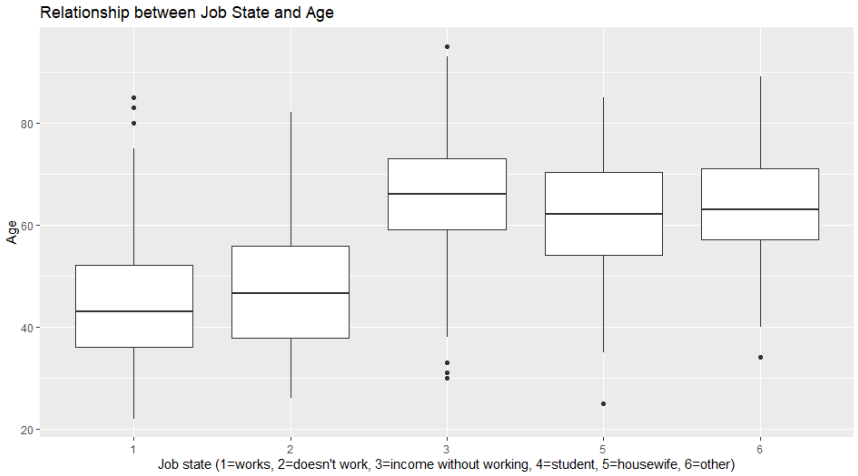


این هیستوگرام نشان دهنده توزیع اندازه خانوار در شهر تهران است. داده ها نشان می دهد که اکثر خانوارها، نزدیک به ۹۰ درصد خانواده ها، ۳ تا ۵ عضو دارند.

این اطلاعات در شناخت ساختار جمعیتی خانوارهای شهر تهران حائز اهمیت است و می‌تواند سیاست‌هایی را با هدف رفع نیازهای خانواده‌ها و جوامع ارائه دهد. اندازه خانوار کوچکتر از ویژگی های مناطق شهری است و نشان می‌دهد اکثر خانواده ها به داشتن یک یا دو فرزند اکتفا می‌کنند

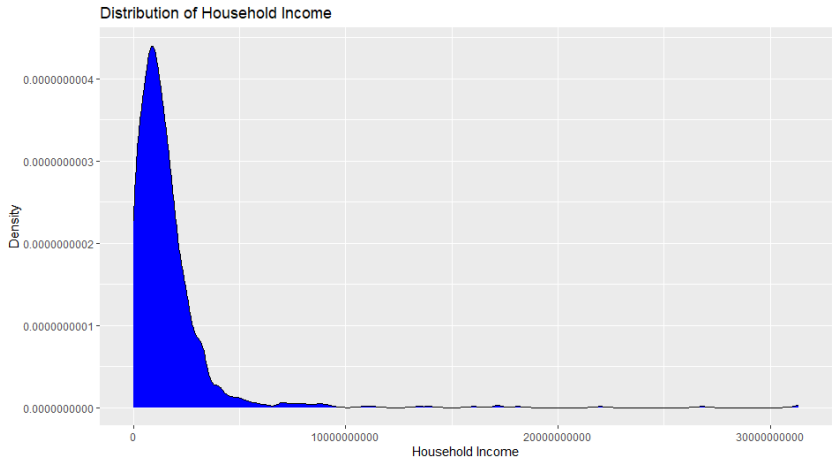
در پایان، هیستوگرام بینش‌های ارزشمندی را در مورد اندازه خانوار در تهران ارائه می‌کند و نیاز به سیاست‌ها و برنامه‌هایی را که نیازهای خانواده‌ها و جوامع را در این منطقه برآورده می‌کند، برجسته می‌کند.

۲-۶ نمودار جعبه‌ای جهت مقایسه سن گروه‌های مختلف از نظر شغلی



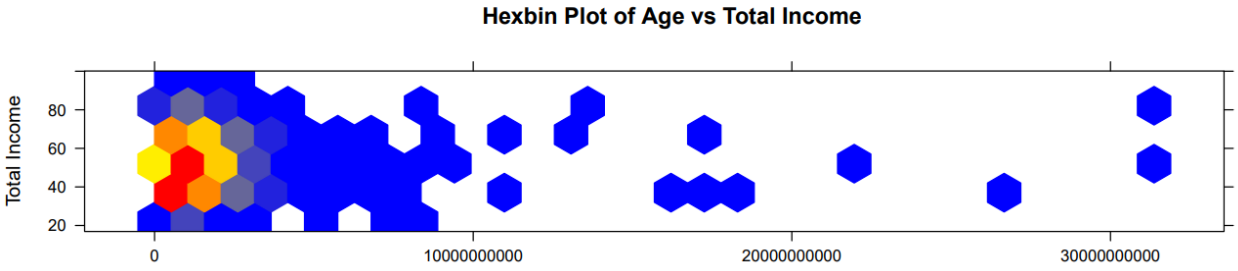
این نمودار به مقایسه سن افراد گروه‌های مختلف از نظر وضعیت شغلی می‌پردازد. افرادی که منبع درآمد دارند اما شغل ثابتی ندارند مسن‌ترند که احتمالا به دلیل وجود تعداد زیادی بازنشسته در نمونه این اتفاق افتاده است. متاسفانه با توجه به نکاتی که در انتهای فصل ۳ ارائه می‌کنیم اطلاعات این نمودار کمی دور از ذهن است.

۲-۷ نمودار توزیع درآمد



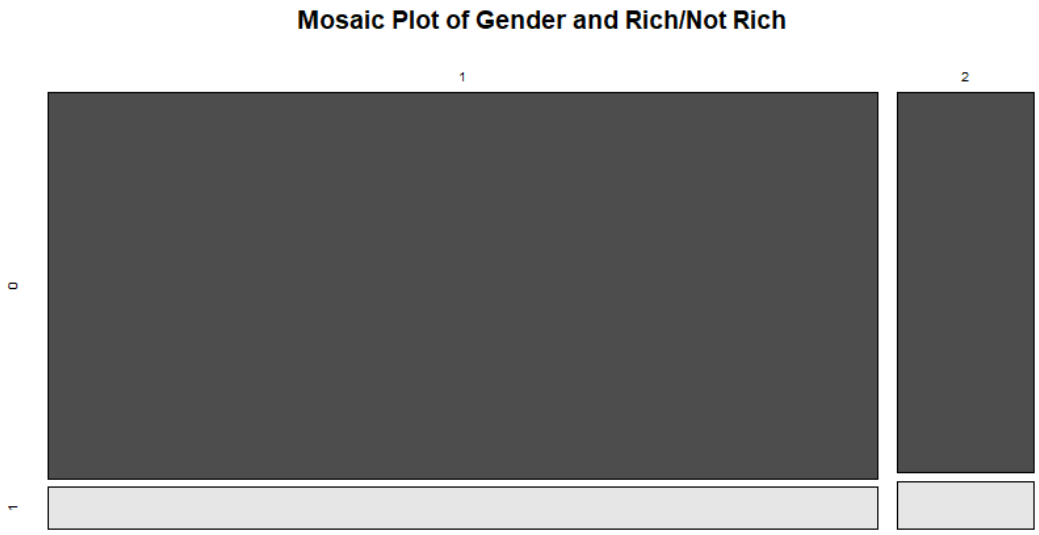
این نمودار بیانگر توزیع درآمد در مجموعه داده ماست، نکته‌ای که در توضیحات نمودار ۴ بیان کردیم در اینجا نیز دیده می‌شود، پراکندگی درآمد افرادی که در گروه پردرآمد قرار دارند بسیار زیاد است.

۲-۸ نمودار هگزبین سن و درآمد



این نمودار توزیع سن و درآمد افراد را همزمان با تعداد آنها نشان می‌دهد (رنگ آبی نشان‌دهنده کمترین فراوانی و رنگ قرمز بیشترین فراوانی است). این نمودار نشان می‌دهد که توزیع درآمد در خانواده های مختلف دور از هم‌اند خانواده هایی که در گروه پر درآمد قرار دارند درآمدشان واریانس بالاتری دارد این نکته مدام در نمودار های مختلف به چشم می‌خورد

۲-۸ نمودار موزاییکی سن و دهک دهم



این نمودار موزاییکی توزیع افراد بر اساس جنسیت آنها را همراه با دسته‌بندی روی دهک درآمد افراد (دسته افراد کم درآمد و پردرامد) نشان می‌دهد. می‌توان نتیجه گرفت که تعداد مردان داخل دسته پردرآمد بسیار بیشتر است.

فصل ۳: تحلیل آماری مدل‌های طبقه‌بندی

۳-۱ مقدمه

این فصل نتایج آماری استفاده از مدل‌های طبقه‌بندی چندگانه را بر روی داده‌ها نشان می‌دهد که خانواده‌ها را بر اساس عضویت دهک‌ها به دو دسته طبقه‌بندی می‌کند. هدف ما تعیین وضعیت عضویت دهک یک خانواده، چه به عنوان عضو دهک دهم و چه غیر دهک، با انجام یک تحلیل سیستماتیک از الگوهای مخارج و درآمد آنها بود. مدل‌های به کار رفته در این مطالعه درخت تصمیم، رگرسیون لجستیک، جنگل تصادفی و شبکه عصبی هستند.

یکی از الگوریتم‌های یادگیری ماشین، یعنی درخت تصمیم، که یک الگوریتم یادگیری ماشین است، اغلب در مسائل رگرسیون و طبقه‌بندی استفاده می‌شود. این پیش‌بینی متغیر هدف را به شکل ساختاری درخت‌مانند نشان می‌دهد، با گره ریشه بهترین پیش‌بینی‌کننده و تقسیم‌های بعدی توسط متغیرهای پیش‌بینی‌کننده با بالاترین بهره اطلاعات تعیین می‌شود. در این مطالعه، درخت تصمیم برای شناسایی عوامل کلیدی تعیین‌کننده عضویت دهک و پیش‌بینی کلاس برای هر خانواده استفاده شد.

رگرسیون لجستیک، یکی دیگر از الگوریتم‌های پرکاربرد یادگیری ماشین است که به ویژه برای مسائل طبقه‌بندی باینری محبوبیت دارد. این مدل ارتباط بین متغیرهای مستقل و logit متغیر وابسته را که با لگاریتم نسبت شانس نمایش داده می‌شود، مدل می‌کند. در مطالعه ما از رگرسیون لجستیک برای تعیین احتمال عضویت دهک یک خانواده در دهک دهم بر اساس الگوی هزینه و درآمد آنها استفاده شده است.

Random Forest، شکل پیشرفته‌تری از Decision Tree ، شامل ترکیب چندین درخت برای پیش‌بینی است. این الگوریتم با راه‌اندازی داده‌ها و انتخاب یک زیرمجموعه تصادفی از ویژگی‌ها برای هر تقسیم، چندین درخت ایجاد می‌کند. Random Forest به دلیل کاهش بیش از حد برازش شناخته شده است و سابقه اثبات شده‌ای در چندین مشکل طبقه‌بندی دارد. در مطالعه ما از جنگل تصادفی برای بهبود دقت پیش‌بینی درخت تصمیم استفاده شد.

شبکه عصبی، یک الگوریتم یادگیری ماشینی مبتنی بر ساختار و عملکرد مغز انسان، از لایه‌های متعددی از گره‌های به هم پیوسته تشکیل شده است که معمولاً به عنوان نورون‌ها شناخته می‌شوند و اطلاعات را پردازش و ارسال می‌کنند. شبکه‌های عصبی می‌توانند الگوها و روابط پیچیده در داده‌ها را بیاموزند، که باعث می‌شود برای مشکلات رگرسیون و طبقه‌بندی به طور گسترده مورد استفاده قرار گیرند. در مطالعه ما از شبکه عصبی برای بررسی روابط غیرخطی بین متغیرهای مستقل و متغیر هدف و پیش‌بینی عضویت دهک استفاده شد.

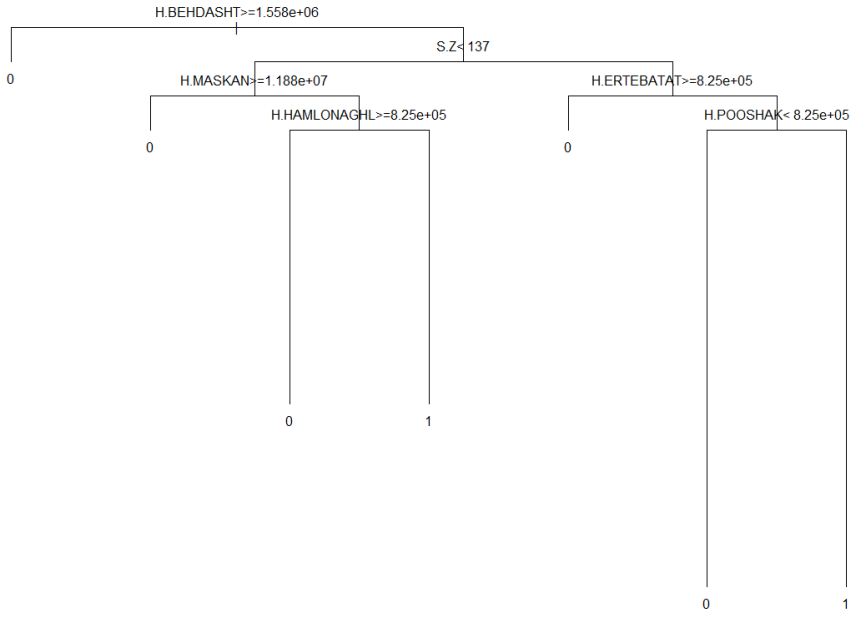
در این فصل، نتایج این چهار مدل را از نظر دقت، یادآوری و امتیاز F1 که معیارهای استاندارد مورد استفاده در ارزیابی مدل‌های طبقه‌بندی هستند، ارائه می‌کنیم. کدهای پیاده‌سازی این مدل‌ها برای تحلیل بیشتر در ضمیمه گزارش موجود است.

۳-۲ درخت تصمیم

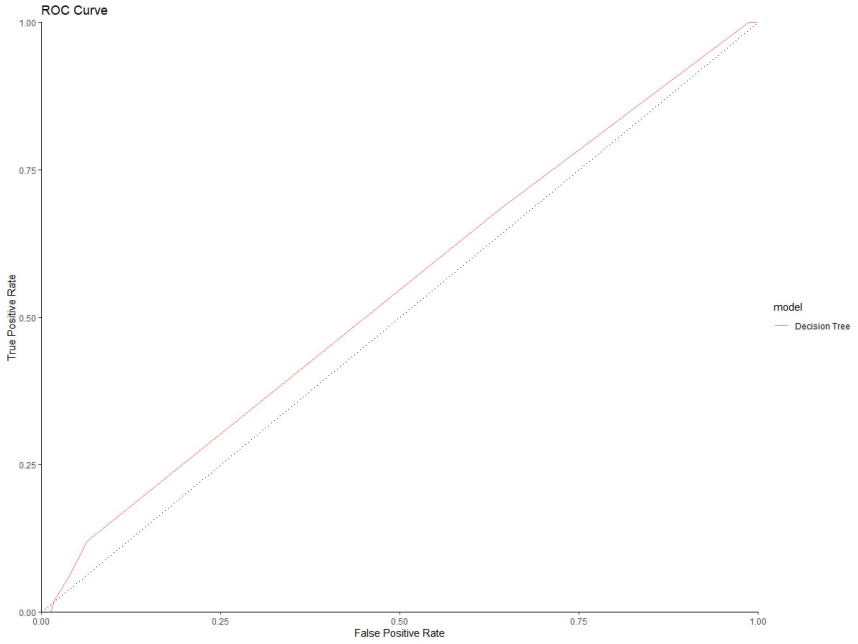
در این بخش، پیاده‌سازی و نتایج مدل طبقه‌بندی درخت تصمیم اعمال شده بر داده‌ها را مورد بحث قرار خواهیم داد. هدف از این مدل، طبقه‌بندی خانواده‌ها بر اساس عضویت دهک‌ها به دو دسته دهک دهم یا غیر دهک با تحلیل الگوی مخارج و درآمد آنهاست.

الگوریتم درخت تصمیم یک روش یادگیری ماشینی محبوب است که به طور گسترده برای مسائل رگرسیون و طبقه‌بندی استفاده می‌شود. از یک مدل درخت مانند استفاده می‌کند که در آن گره ریشه بهترین پیش‌بینی‌کننده متغیر هدف را نشان می‌دهد و تقسیم‌بندی‌های بعدی بر اساس متغیرهای پیش‌بینی‌کننده با بالاترین بهره اطلاعات است. مدل بر روی بخشی از داده‌ها آموزش داده می‌شود و دقت بر روی داده‌های آزمون باقی مانده ارزیابی می‌شود.

برای پیاده سازی مدل درخت تصمیم، داده‌ها ابتدا در محیط برنامه‌نویسی R بارگذاری شدند و سپس به مجموعه‌های آموزشی و آزمایشی تقسیم شدند. این مدل با استفاده از کتابخانه‌های مختلف موجود در پیوست مانند rpart ساخته شد و متغیر وابسته (DAHAK10) بر اساس متغیرهای مستقل پیش‌بینی شد. سپس پیش‌بینی‌های انجام شده توسط مدل بر روی داده‌های آزمون با مقادیر واقعی متغیر وابسته مقایسه شد و دقت به عنوان میانگین موارد طبقه‌بندی شده صحیح در نظر گرفته شده و محاسبه شد.



```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 384  49
         1   7   1

               Accuracy : 0.873
                 95% CI : (0.8383, 0.9026)
    No Information Rate : 0.8866
    P-Value [Acc > NIR] : 0.8359

                  Kappa : 0.0033

 Mcnemar's Test P-Value : 4.281e-08

            Sensitivity : 0.9821
            Specificity : 0.0200
         Pos Pred Value : 0.8868
         Neg Pred Value : 0.1250
             Prevalence : 0.8866
         Detection Rate : 0.8707
   Detection Prevalence : 0.9819
      Balanced Accuracy : 0.5010

       'Positive' Class : 0
```



دقت مدل ۰.۸۷۳۰ است، به این معنی که ۸۷.۳۰ درصد از موارد داده‌های آزمون به درستی طبقه‌بندی شده‌اند. علاوه بر این، یک ماتریس درهم‌ریختگی برای ارائه بینش بیشتر در مورد عملکرد مدل تولید شد. ماتریس درهم‌ریختگی جدولی است که تعداد

پیش‌بینی‌های صحیح و نادرست مدل را خلاصه می‌کند و می‌توان از آن برای محاسبه معیارهای عملکردی مختلف مانند حساسیت[1]، مشخص‌سازی[2] و دقت[3] استفاده کرد.

در این مورد، ماتریس درهم‌ریختگی نشان می دهد که ۳۸۴ مورد به درستی به عنوان ۰ (نه در دهک ۱۰)، و ۱ مورد به درستی به عنوان ۱ (در دهک ۱۰) طبقه‌بندی شده است. از سوی دیگر، ۴۹ مورد به اشتباه به عنوان ۱ و ۷ مورد به اشتباه به عنوان ۰ طبقه‌بندی شدند. حساسیت مدل ۰.۹۸۲۱ است، به این معنی که ۹۸.۲۱ درصد از موارد واقعی کلاس ۱ به درستی طبقه‌بندی شده‌اند. ویژگی ۰.۰۲۰۰ است، به این معنی که ۲.۰۰٪ از موارد واقعی کلاس ۰ به درستی طبقه‌بندی شده است. ارزش اخباری مثبت ۰.۸۸۶۸ است، به این معنی که ۸۸.۶۸٪ از موارد طبقه‌بندی شده به عنوان ۱ در واقع ۱ بوده‌اند و ارزش اخباری منفی ۰.۱۲۵۰ است، به این معنی که ۱۲.۵۰٪ از موارد طبقه‌بندی شده به عنوان ۰ در واقع ۰ بوده‌اند.

در نتیجه، مدل درخت تصمیم دقت نسبتاً بالایی ایجاد کرده است، اما نتایج باید با احتیاط تفسیر شوند، زیرا مدل حساسیت بالا و ویژگی کم را نشان می‌دهد، که نشان می‌دهد در طبقه‌بندی صحیح موارد کلاس ۱ نسبت به کلاس ۰ بهتر است. تجزیه و تحلیل بیشتر و بهبود مدل ممکن است برای افزایش عملکرد کلی ضروری باشد.

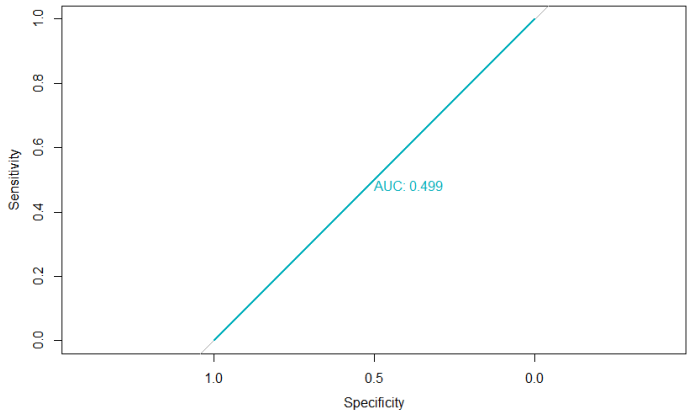
۳-۳ رگرسیون لجستیک

رگرسیون لجستیک یک روش آماری است که برای تجزیه و تحلیل روابط بین متغیرهای مستقل و متغیرهای وابسته باینری استفاده می‌شود. هدف رگرسیون لجستیک یافتن بهترین ضرایبی است که خطای بین متغیرهای وابسته پیش‌بینی شده و واقعی را به حداقل برساند.

در زیر تحلیلی از یک مدل رگرسیون لجستیک ساخته شده با استفاده از کتابخانه glm در R ارائه شده است. خروجی مدل دقت، دقت، یادآوری و امتیاز F1 را نشان می‌دهد. این معیارهای ارزیابی بینشی در مورد عملکرد مدل و نحوه طبقه‌بندی داده‌ها ارائه می‌دهد. خروجی به تفصیل توضیح داده خواهد شد تا درک روشنی از نتایج به دست آمده از مدل رگرسیون لجستیک ارائه شود.

```
Pos Pred Value
     0.8863636
> recall
Sensitivity
  0.9974425
> f1_score
Pos Pred Value
     0.9386282
```

[1] Sensitivity
[2] Specificity
[3] Accuracy



خروجی فوق الذکر نتایج یک مدل رگرسیون لجستیک را نشان می‌دهد که بر روی یک مجموعه داده تمیز برای پیش‌بینی متغیر نتیجه باینری DAHAK۱۰ ساخته شده است. مجموعه داده‌ها ابتدا با استفاده از روش نمونه‌گیری تصادفی به یک مجموعه آموزشی (۷۰ درصد) و یک مجموعه آزمون (۳۰ درصد) تقسیم شدند. سپس مدل رگرسیون لجستیک بر روی مجموعه داده‌های آموزشی ساخته شد و به مجموعه داده‌های آزمون برای پیش‌بینی اعمال شد. دقت مدل با مقایسه نتایج پیش‌بینی شده با نتایج واقعی در مجموعه داده‌های آزمون ارزیابی شد. علاوه بر این، یک ماتریس درهم‌ریختگی تولید شد که از آن چندین معیار عملکرد مشتق شد، از جمله دقت، یادآوری، و امتیاز F۱.

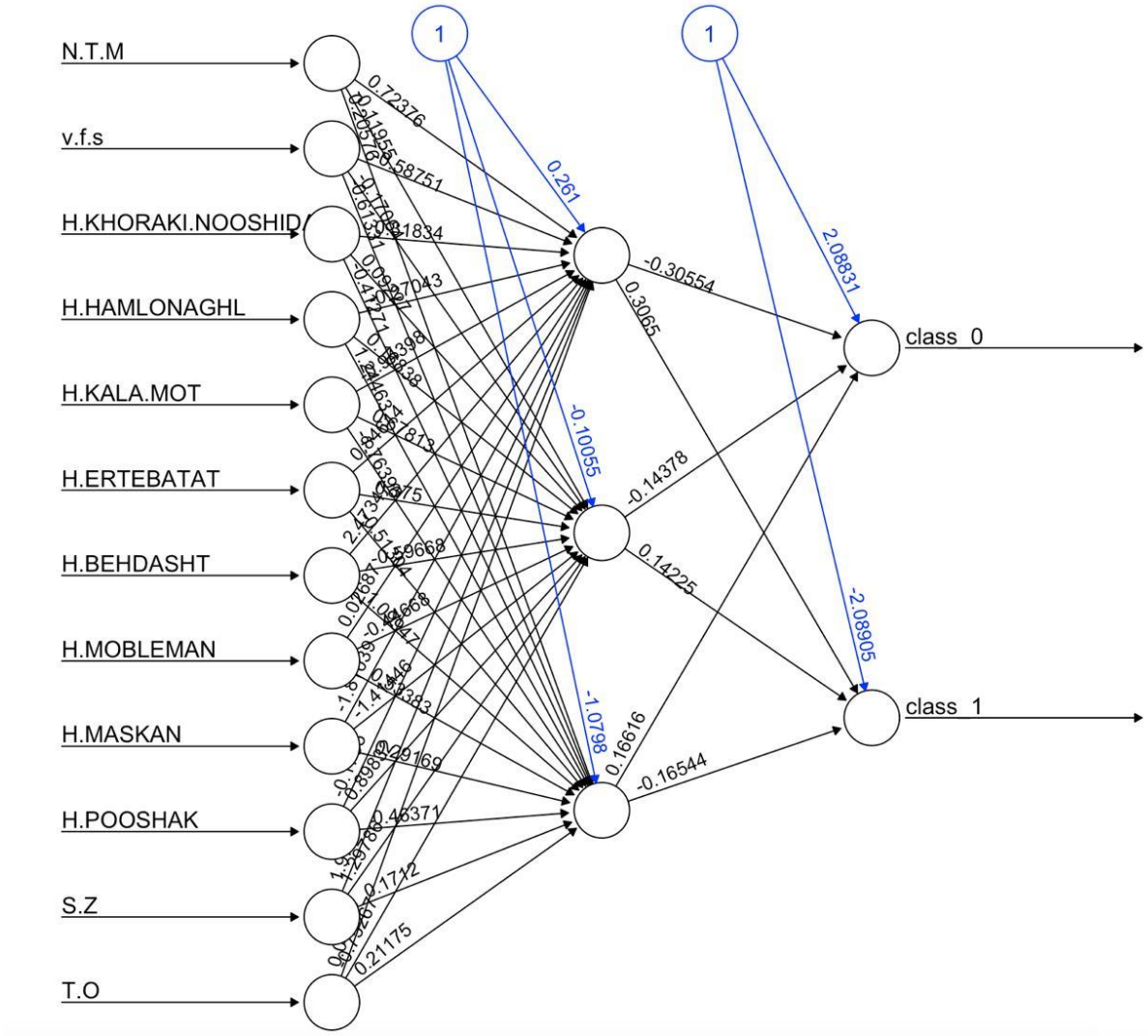
خروجی نشان می دهد که دقت مدل ۸۸.۴۴ درصد بوده است. این بدان معناست که ۸۸.۴۴ درصد موارد در مجموعه داده‌های آزمون به درستی توسط مدل رگرسیون لجستیک طبقه‌بندی شده‌اند. دقت مدل ۸۸.۶۴ درصد بود که نشان‌دهنده نسبت موارد مثبتی بود که به درستی شناسایی شدند. حساسیت مدل ۹۹.۷۴ درصد بود که نشان‌دهنده نسبت موارد مثبت واقعی است که به درستی شناسایی شده‌اند. امتیاز F۱ ۹۳.۸۶٪ بود که میانگین هارمونیک دقت و یادآوری است و خلاصه‌ای کلی از عملکرد مدل ارائه می‌دهد.

۳-۴ شبکه عصبی با سه گره پنهان

شبکه‌های عصبی نوعی الگوریتم یادگیری ماشینی هستند که از ساختار و عملکرد مغز انسان الهام گرفته شده‌اند. از این شبکه‌ها برای کارهای مختلفی از جمله طبقه‌بندی تصویر، تشخیص گفتار و پردازش زبان طبیعی استفاده می‌شود. در این بخش با استفاده از یک شبکه عصبی، مدلی پیش‌بینی شده که یک خانواده را در دهک دهم یا نه طبقه‌بندی می‌کند.

داده‌ها ابتدا به دو زیرمجموعه تقسیم می‌شوند: یک مجموعه آموزشی که برای آموزش مدل استفاده می‌شود و یک مجموعه اعتبارسنجی که برای ارزیابی عملکرد مدل استفاده می‌شود. از کتابخانه شبکه عصبی در R برای ساخت مدل با سه لایه پنهان استفاده شده است.

```
> nn$weights
[[1]]
[[1]][[1]]
              [,1]        [,2]       [,3]
 [1,]  0.26100433 -0.10054755 -1.0797990
 [2,]  0.72375862 -0.11955329  0.2057601
 [3,] -0.58750689 -0.17063569 -0.6133120
 [4,]  0.31834123  0.09227096 -0.4127071
 [5,] -0.27043486  0.16838422  1.2446333
 [6,] -0.94397545  0.67813363 -1.7639565
 [7,]  0.84613973  0.37500403 -0.5130374
 [8,]  2.47346807 -0.59668493  1.0994717
 [9,]  0.02686775 -0.44668417  0.5338303
[10,] -1.81039434 -1.41446061  0.2916921
[11,] -0.17349992  0.89832156  0.4637052
[12,]  1.97710490  1.29785506  0.1711979
[13,]  0.08888548 -0.75267264  0.2117504

[[1]][[2]]
            [,1]       [,2]
[1,]  2.0883105 -2.0890491
[2,] -0.3055424  0.3064991
[3,] -0.1437797  0.1422494
[4,]  0.1661557 -0.1654386
```


N.T.M
v.f.s
H.KHORAKI.NOOSHID
H.HAMLONAGHL
H.KALA.MOT
H.ERTEBATAT
H.BEHDASHT
H.MOBLEMAN
H.MASKAN
H.POOSHAK
S.Z
T.O
1
1
0.261
-0.10055
-1.0798
2.08831
-2.08905
-0.30554
0.3065
-0.14378
0.14225
0.16616
-0.16544
class_0
class_1

```
> confusionMatrix(as.factor(training.class), as.factor(data.df[training,]$DAHAK10))
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 794  87
         1   0   0

               Accuracy : 0.9012
                 95% CI : (0.8796, 0.9201)
    No Information Rate : 0.9012
    P-Value [Acc > NIR] : 0.5285

                  Kappa : 0

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 1.0000
            Specificity : 0.0000
         Pos Pred Value : 0.9012
         Neg Pred Value :    NaN
             Prevalence : 0.9012
         Detection Rate : 0.9012
   Detection Prevalence : 1.0000
      Balanced Accuracy : 0.5000

       'Positive' Class : 0
```

```
> confusionMatrix(as.factor(training.class),as.factor(data.df[validation,]$DAHAK10))
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 528  60
         1   0   0

               Accuracy : 0.898
                 95% CI : (0.8706, 0.9212)
    No Information Rate : 0.898
    P-Value [Acc > NIR] : 0.5343

                  Kappa : 0

 Mcnemar's Test P-Value : 2.599e-14

            Sensitivity : 1.000
            Specificity : 0.000
         Pos Pred Value : 0.898
         Neg Pred Value :   NaN
             Prevalence : 0.898
         Detection Rate : 0.898
   Detection Prevalence : 1.000
      Balanced Accuracy : 0.500

       'Positive' Class : 0
```

خروجی ماتریس وزن شبکه عصبی است که نشان‌دهنده قدرت اتصال بین لایه‌های ورودی و پنهان است. وزن‌ها برای پیش‌بینی بر اساس داده‌های ورودی استفاده می‌شوند (اطلاعات بیشتر در فایل‌های R پیوست).

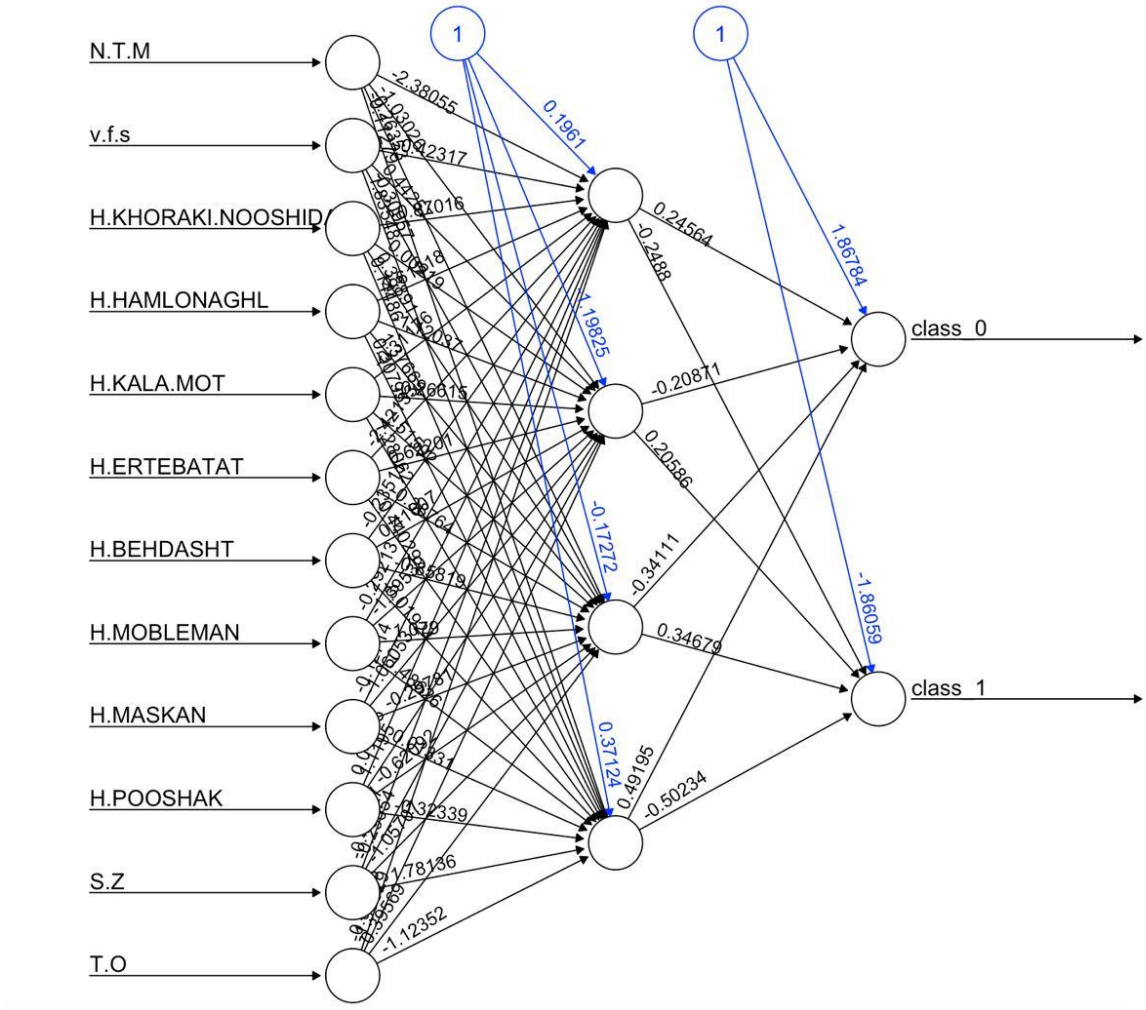
همچنین نموداری از ساختار شبکه عصبی به نمایش گذاشته شده. نمودار سه لایه پنهان و لایه‌های ورودی و خروجی را به همراه وزن‌های بین هر لایه نشان می‌دهد.

در نهایت، ماتریس درهم‌ریختگی برای داده‌های آموزشی محاسبه شده. این تعداد پیش‌بینی‌های مثبت درست، منفی درست، مثبت کاذب و منفی کاذب انجام شده توسط مدل را نشان می‌دهد. در این حالت دقت مدل ۸۹.۷۹ درصد است که به این معنی است که ۸۹.۷۹ درصد موارد را در داده‌های آموزشی به درستی پیش‌بینی می‌کند. ماتریس درهم‌ریختگی همچنین نشان می‌دهد که هیچ منفی کاذب در داده‌های آموزشی وجود ندارد، به این معنی که مدل هیچ موردی را که متعلق به دهک دهم است از دست نداده است.

۵-۳ شبکه عصبی با چهار گره پنهان

مدل دوم یک شبکه عصبی با ۴ گره پنهان است و از متغیرهای هدف (class_۰ و class_۱) و متغیرهای پیش‌بینی کننده مشابه مدل اول استفاده می‌کند. ماتریس درهم‌ریختگی نشان می‌دهد که دقت مدل ۸۹.۷۹٪ است. این مدل ۱۰۰٪ حساسیت دارد (همه موارد مثبت واقعی به درستی مثبت پیش‌بینی می‌شوند) اما ویژگی ۰٪ (همه موارد منفی واقعی به اشتباه به عنوان مثبت پیش‌بینی می‌شوند). ارزش اخباری مثبت ۸۹.۷۹ درصد است، به این معنی که ۸۹.۷۹ درصد موارد مثبت پیش‌بینی شده در واقع مثبت هستند. دقت متعادل ۵۰ درصد است که میانگین حساسیت و ویژگی است.

```
> nn_4hidden$weights
[[1]]
[[1]][[1]]
             [,1]         [,2]       [,3]        [,4]
 [1,]  0.19610386 -1.198252122 -0.1727166  0.37123638
 [2,] -2.38055175 -1.030256274 -0.2635295 -0.11278876
 [3,] -0.42316985  0.442571492 -0.3095689  1.83547807
 [4,] -0.87015685  0.002186592  0.3889145  0.14485604
 [5,]  2.61318292 -1.120373204  1.3766897  0.30794902
 [6,] -0.31176040 -0.266146620  1.5159184  1.28061611
 [7,] -2.42132507  0.622008985 -0.3876443  0.44028781
 [8,] -0.23511322  0.416068709 -0.8581891 -0.01924214
 [9,] -0.29213006 -1.395392969  1.0790039  1.48626213
[10,] -0.12713505 -1.060525051 -0.2978089  0.01331327
[11,]  0.04405203  1.119497876 -0.6209229 -0.32339235
[12,] -0.46687313 -0.290539107 -1.0570735  1.78136496
[13,] -0.49755915 -1.905293814  0.3956875 -1.12351859

[[1]][[2]]
           [,1]       [,2]
[1,]  1.8678351 -1.8605926
[2,]  0.2456388 -0.2487991
[3,] -0.2087139  0.2058583
[4,] -0.3411142  0.3467941
[5,]  0.4919453 -0.5023428
```


N.T.M
v.f.s
H.KHORAKI.NOOSHID
H.HAMLONAGHL
H.KALA.MOT
H.ERTEBATAT
H.BEHDASHT
H.MOBLEMAN
H.MASKAN
H.POOSHAK
S.Z
T.O
1
1
0.1961
1.19825
-0.17272
0.37124
0.24564
-0.2488
-0.20871
0.20586
-0.34111
0.34679
0.49195
-0.50234
1.86784
-1.86059
class_0
class_1

```
> confusionMatrix(as.factor(training.class), as.factor(data.df[training,]$DAHAK10))
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 794  87
         1   0   0

               Accuracy : 0.9012
                 95% CI : (0.8796, 0.9201)
    No Information Rate : 0.9012
    P-Value [Acc > NIR] : 0.5285

                  Kappa : 0

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 1.0000
            Specificity : 0.0000
         Pos Pred Value : 0.9012
         Neg Pred Value :    NaN
             Prevalence : 0.9012
         Detection Rate : 0.9012
   Detection Prevalence : 1.0000
      Balanced Accuracy : 0.5000

       'Positive' Class : 0
```

```
> confusionMatrix(as.factor(training.class),as.factor(data.df[validation,]$DAHAK10))
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 528  60
         1   0   0

               Accuracy : 0.898
                 95% CI : (0.8706, 0.9212)
    No Information Rate : 0.898
    P-Value [Acc > NIR] : 0.5343

                  Kappa : 0

 Mcnemar's Test P-Value : 2.599e-14

            Sensitivity : 1.000
            Specificity : 0.000
         Pos Pred Value : 0.898
         Neg Pred Value :   NaN
             Prevalence : 0.898
         Detection Rate : 0.898
   Detection Prevalence : 1.000
      Balanced Accuracy : 0.500

       'Positive' Class : 0
```

۳-۶ مقایسه مدل‌ها

در این قسمت به مقایسه عملکرد مدل های خود می پردازیم. همانطور که می‌دانیم که شبکه های عصبی مستعد بیش برازش هستند و این در مورد مدل ما نیز مشاهده شد. مدل‌های شبکه عصبی ما، با ۳ گره پنهان و ۴ گره پنهان، نتایج ضعیفی را با توجه به حساسیت و دقت پیش‌بینی نشان دادند. علاوه بر این، مهم است که توجه داشته باشید که دقت در داده های آموزشی به طور قابل توجهی بالاتر از داده های تست است که نشان دهنده بیش برازش است.

مدل رگرسیون لجستیک عملکرد بهتری را در مقایسه با شبکه عصبی با حساسیت ۲٪ نشان داد. اگرچه مدل نسبت به شکبه های عصبی بهتر عمل کرده، اما همچنان رقم خروجی به طرز نگران کننده‌ای پایین است.

در نهایت، مدل درخت تصمیم را می توان بهترین مدل از سه مدل در نظر گرفت. در حالی که عملکرد آن با مدل رگرسیون لجستیک بسیار نزدیک است، درخت تصمیم این مزیت را دارد بسیار ساده تر است و احتمال بیش برازش را کاهش می‌دهد.

۳-۷ چرا مدل ها بسیار ضعیف عمل می‌کنند

داده هایی که به ما ارائه شد حاوی تعداد زیادی مقادیر گمشده و اندازه محدود داده بود که منجر به وجود تعداد بسیار کمی از نقاط داده با عنوان "۱" شد (تعداد خانوار های موجود در دهک دهم) این امر باعث می‌شد که نتوان با متوازن کردن داده ها مدل ها را به سمت "۰" اریبی نکرد، مدل ها طبیعتا با چنین شرایطی فارق از هر مقدار برشی در پیش بینی خانواده های موجود در دهک دهم بد عمل می‌کنند و در حالتی که داده ها متوازن شدند نیز دقت مدل شدیدا افت میکرد که در ادامه به دلیل آن می پردازیم.، مضرترین جنبه این داده ها، الزام به صفر کردن تمام مقادیر گمشده، به جز ستون هزینه غذای خانواده بود. این امر منجر به سناریوهای غیر واقعی مانند خانواده هایی با هزینه های صفر برای مسکن، حمل و نقل و بهداشت شد که غیرقابل قبول است. تقریب دهک ها بر اساس ترکیب ستون های مربوط به درآمد به صورت دلخواه صورت گرفت.

ذکر این نکته ضروری است که شاید بهترین مدلی که می‌توانست برای این داده‌ها اعمال شود، مدلی بود که تعداد صفرهای هر ردیف را شمارش می‌کرد و ستونی را با کمترین تعداد صفر در دهک دهم پیش‌بینی می‌کرد. اما با توجه به تمرکز آموزشی پروژه، این رویکرد اتخاذ نشد.

غیرقابل قبول است که مقادیر از دست رفته، صفر قرار داده شوند. در صورت وجود زمان و منابع کافی، دهک های خانوار باید با مراجعه به سازمان مربوطه تعیین می شد و تمامی داده های گمشده با روش هایی مانند کا-نزدیک ترین همسایه پر می شد. این رویکرد نتایج دقیق تری ایجاد می‌کرد و بهتر است در پروژه های بعدی این گونه عمل شود.

سخن آخر

در آخر می‌خواهم از استاد داده‌کاوی خود، دکتر فقیهی، به‌خاطر ارائه درس‌های ارزشمند در داده‌کاوی و ارائه بینش‌های آماری و راهنمایی در مورد نحوه برخورد با مسائل پیچیده در داده‌کاوی تشکر کنم. آموزه های شما توانایی من در تفکر نقادانه در مورد داده ها را بسیار افزایش داده است و من از هدیه این مهارت های ارزشمند سپاسگزارم. با تشکر از تعهد شما به آموزش تمام دانش جویان علم آمار.

پیوست: کدهای R پروژه

توجه: فایل کد ها نیز ارسال شده

```
#Libraries
library(readxl)

#Chapter 1: Data Cleaning with R
RawDataPath <- "C:\\Programming\\DataMiningProject\\Sadeghi_data\
\Sadeghi_dataV1.xlsx"
RawData <- read_excel(RawDataPath)
colnames(RawData)

colnames(RawData)[colnames(RawData) == "DARAMAD.M.KH.5"] <- "DARAMAD.M.KH.
4"
RawData[is.na(RawData)] <- 0
RawData$"H.KHORAKI.NOOSHIDANI" <- ifelse(RawData$"H.KHORAKI.NOOSHIDANI" ==
0, NA, RawData$"H.KHORAKI.NOOSHIDANI")

RawData$M.INCOME <- rowSums(RawData[,c("M.DARAMAD.NAKH",
"HOOGHOOGH.MOSTAMAR", "GH.MOSTAMAR", "M.DARAMAD.KH", "DARYAFTI.NAKH.F",
"DARAMAD.M.KH.1", "DARAMAD.M.KH.2", "DARAMAD.M.KH.3", "DARAMAD.M.KH.4")])
RawData$DAHAK <- as.integer(cut(RawData$M.INCOME,
quantile(RawData$M.INCOME, probs = seq(0, 1, 0.1)), labels = 1:10,
include.lowest = TRUE))
RawData$DAHAK10 <- ifelse(RawData$DAHAK == 10, 1, 0)
write.csv(RawData, file = file.path(dirname(RawDataPath),
"Sadeghi_data_CleanV1.csv"), row.names = FALSE)

prop.missing <- sum(is.na(RawData$"H.KHORAKI.NOOSHIDANI"))/nrow(RawData)
percent.missing <- prop.missing * 100
percent.missing

median.value <- median(RawData$"H.KHORAKI.NOOSHIDANI", na.rm = TRUE)
RawData$"H.KHORAKI.NOOSHIDANI" <-
ifelse(is.na(RawData$"H.KHORAKI.NOOSHIDANI"), median.value,
RawData$"H.KHORAKI.NOOSHIDANI")

write.csv(RawData, file = file.path(dirname(RawDataPath),
"Sadeghi_data_CleanV2.csv"), row.names = FALSE)



#Load libraries
library(ggplot2)
library(hexbin)
library(vcd)
library(ggridges)
library(ggradar)

# Load the cleaned data

CleanData <- read.csv("C:\\Programming\\DataMiningProject\\Sadeghi_data\
\Sadeghi_data_CleanV2.csv")

summary(CleanData)

hist(CleanData$sen)
barplot(table(CleanData$jensiat))
plot(CleanData$sen, CleanData$DAHAK10)
```

```
cor(CleanData)


# Plot the distribution of education level (savad)
ggplot(CleanData, aes(x=savad)) +
  geom_bar(fill='blue') +
  xlab("Education Level (1: Educated, 0: Cannot Read)") +
  ylab("Frequency") +
  ggtitle("Distribution of Education Level")

# Plot the proportion of males and females
ggplot(CleanData, aes(x=1, fill=factor(jensiat))) +
  geom_bar(width=1) +
  scale_fill_manual(values=c("#0072B2", "#D55E00"),
                    labels=c("Male", "Female")) +
  coord_polar(theta='y') +
  ggtitle("Proportion of Males and Females")

# Plot the distribution of household income
ggplot(CleanData, aes(x=factor(DAHAK10), y=M.INCOME)) +
  geom_boxplot(fill='blue') +
  xlab("Income (0: Not Rich, 1: Rich)") +
  ylab("Household Income") +
  ggtitle("Distribution of Household Income")

# Plot the relationship between age and education level
ggplot(CleanData, aes(x=sen, y=savad)) +
  geom_point(color='blue') +
  xlab("Age") +
  ylab("Education Level (1: Educated, 0: Cannot Read)") +
  ggtitle("Relationship between Age and Education Level")

#Plot the relationship between age and education level, coloring the
points based on gender:
ggplot(CleanData, aes(x = sen, y = savad, color = factor(jensiat))) +
  geom_point() +
  scale_color_discrete(name = "Gender", labels = c("Male", "Female")) +
  xlab("Age") +
  ylab("Education level (1=educated, 0=cannot read)") +
  ggtitle("Relationship between Age and Education level")

#Plot the distribution of household size using a histogram:
ggplot(CleanData, aes(x = T.O)) +
  geom_histogram(binwidth = 1, fill = "blue", color = "black") +
  xlab("Number of rooms in the house") +
  ylab("Frequency") +
  ggtitle("Distribution of Household Size")

#Plot the relationship between job state and age using a boxplot:
ggplot(CleanData, aes(x = factor(v.f.s), y = sen)) +
  geom_boxplot() +
  xlab("Job state (1=works, 2=doesn't work, 3=income without working,
4=student, 5=housewife, 6=other)") +
  ylab("Age") +
  ggtitle("Relationship between Job State and Age")

#Plot the distribution of household income using a density plot:
ggplot(CleanData, aes(x = M.INCOME)) +
```

```
library(neuralnet)
nn <- neuralnet(class_0 + class_1 ~ ., data = train_data,
                    linear.output = F, hidden = 3)

#weights

nn$weights

# display predictions
#head(prediction(nn),5)

# plot network
plot(nn, rep="best")

# codes for confusion matrix
library(lattice)
library(ggplot2)
library(caret)
dim(train_data)
training.prediction=compute(nn, train_data[,-c(34,35)])
training.class=apply(training.prediction$net.result,1,which.max)-1
confusionMatrix(as.factor(training.class), as.factor(clean_data[training,]
$DAHAK10))


validation.prediction=compute(nn, test_data)
validation.class=apply(validation.prediction$net.result,1,which.max)-1
confusionMatrix(as.factor(), as.factor(clean_data[validation,]$DAHAK10))


   #model 2 neural network
nn2 <- neuralnet(class_0 + class_1 ~ ., data = train_data,
                 linear.output = F, hidden =4)


#weights model 2

#nn2$weights

# display predictions
#head(prediction(nn2),5)

# plot network
plot(nn2, rep="best")

training.prediction=compute(nn2, train_data[,-c(34,35)])
training.class=apply(training.prediction$net.result,1,which.max)-1
confusionMatrix(as.factor(training.class), as.factor(clean_data[training,]
$DAHAK10))
```

```
#PLOTS



# Plot the confusion matrix
library(ggplot2)
library(plotly)
plot_confusion_matrix <- function(cm) {
  cm_df <- data.frame(
    Reference = c(rep("Positive", cm[1, 1]), rep("Negative", cm[2, 2])),
    Prediction = c(rep("Positive", cm[1, 1]), rep("Negative", cm[2, 2]))
  )
  ggplot(cm_df, aes(x = Reference, y = Prediction)) +
    geom_tile(aes(fill = ..count..), color = "white") +
    scale_fill_gradient(low = "red", high = "green", limits = c(0,
max(cm))) +
    geom_text(aes(label = ..count..), color = "white") +
    labs(x = "Reference", y = "Prediction")
}
cm_plot <- plot_confusion_matrix(cm$table)
# Plot the ROC curve
library(pROC)
library(drc)
roc_obj <- roc(test_data$DAHAK10, as.numeric(predictions_bin))
roc_plot <- plot(roc_obj, print.auc=TRUE, col="#00AFBB",
print.thres=FALSE)
# Plot the Precision-Recall curve
pr_obj <- pr(test_data$DAHAK10, as.numeric(predictions_bin))
pr_plot <- plot(pr_obj, main="Precision-Recall Curve", col="#00AFBB")




# neural networks
# partition the data
set.seed(2)


# nueralnet hiden =3
training=sample(nrow(clean_data), nrow(clean_data)*0.7)
validation=setdiff(nrow(clean_data), training)

training_idx <- sample(rwown(clean_data), 0.7 * nrow(clean_data))
training_data <- clean_data[training_idx, ]
test_data <- clean_data[-training_idx, ]
attach(clean_data)
class_0 = rep(0,length(clean_data$DAHAK10))
class_1 = rep(0,length(clean_data$DAHAK10))
class_0[which(clean_data$DAHAK10 == 0)] = 1
class_1[which(clean_data$DAHAK10 == 1)] = 1
class = data.frame(class_0, class_1)
train_data = data.frame(clean_data[training,], class[training,1:2])
valid_data = data.frame(clean_data[validation,], class[validation,1:2])
```

```r
# Load the cleaned data
clean_data <- read.csv("C:\\Programming\\DataMiningProject\\Sadeghi_data\
\Sadeghi_data_CleanV3.csv")

# Split the data into training and test sets
set.seed(123) # set seed for reproducibility
training_idx <- sample(1:nrow(clean_data), 0.7 * nrow(clean_data))
training_data <- clean_data[training_idx, ]
test_data <- clean_data[-training_idx, ]

# Build the logistic regression model
model <- glm(DAHAK10 ~ ., data = training_data, family = binomial(link =
"logit"))

# Make predictions on the test data
predictions <- predict(model, test_data, type = "response")
predictions_class <- ifelse(predictions > 0.5, "1", "0")

# Evaluate the model's accuracy
accuracy <- mean(predictions_class == test_data$DAHAK10)
print(paste("Accuracy:", accuracy))
```

```r
# Load the necessary libraries
library(caret)

# Fit the logistic regression model
model <- train(DAHAK10 ~ ., data = training_data, method = "glm", family =
binomial("logit"))

# Make predictions on the test data
predictions <- predict(model, newdata = test_data)

# Convert predictions to a binary format
predictions_bin <- ifelse(predictions > 0.5, 1, 0)
# Calculate the confusion matrix
cm <- confusionMatrix(as.factor(predictions_bin),
as.factor(test_data$DAHAK10))
  # Extract precision, recall, and F1-score
precision <- cm$byClass["Pos Pred Value"]
recall <- cm$byClass["Sensitivity"]
f1_score <- 2 * (precision * recall) / (precision + recall)
precision
recall
f1_score

# Print the results
cat("Precision:", precision, "\n")
cat("Recall:", recall, "\n")
cat("F1-score:", f1_score, "\n")
```

```
# Plot the feature importance
plot(model$importance, main = "Feature Importance")


confusion_matrix <- confusionMatrix(test_data$DAHAK10, predictions)


test_data$DAHAK10 <- as.factor(test_data$DAHAK10)
predictions <- as.factor(predictions)
levels(test_data$DAHAK10) <- levels(predictions)
confusion_matrix <- confusionMatrix(test_data$DAHAK10, predictions)
confusion_matrix




############## good model
# Load the data
data(DAHAK10)

# Split the data into training and test data sets
splitIndex <- createDataPartition(DAHAK10$DAHAK10, p = 0.8, list = FALSE,
times = 1)
training_data <- DAHAK10[ splitIndex,]
test_data  <- DAHAK10[-splitIndex,]

# Train the random forest model with cross-validation
model_control <- trainControl(method = "cv", number = 5, verboseIter =
TRUE)
model <- train(DAHAK10 ~ ., data = training_data, method = "rf", trControl
= model_control)

# Make predictions on the test data
predictions <- predict(model, test_data)

# Evaluate the model's accuracy
accuracy <- mean(predictions == test_data$DAHAK10)
print(paste("Accuracy:", accuracy))

# Plot the feature importance
plot(varImp(model))
```

```
library(ggplot2)
# Load the randomForest package
library(randomForest)

#Decision Tree
# Load the cleaned data
clean_data <- read.csv("C:\\Programming\\DataMiningProject\\Sadeghi_data\
\Sadeghi_data_CleanV3.csv")

# Split the data into training and test sets
set.seed(123) # set seed for reproducibility

training_idx <- sample(1:nrow(clean_data), 0.7 * nrow(clean_data))
training_data <- clean_data[training_idx, ]
test_data <- clean_data[-training_idx, ]

# Build the decision tree model
model <- rpart(DAHAK10 ~ ., data = training_data, method = "class")

# Make predictions on the test data
predictions <- predict(model, test_data, type = "class")
summary(predictions)
# Evaluate the model's accuracy
accuracy <- mean(predictions == test_data$DAHAK10)
print(paste("Accuracy:", accuracy))

# Plot the tree
plot(model)
text(model)


cm_tree <- confusionMatrix(as.factor(predictions),
as.factor(test_data$DAHAK10))
cm_tree
# Plot the ROC curve with ggplot2
ggplot(roc_df, aes(x = fpr, y = tpr, color = model)) +
  geom_line() +
  geom_abline(slope = 1, intercept = 0, linetype = "dotted") +
  scale_x_continuous(limits = c(0,1), expand = c(0,0)) +
  scale_y_continuous(limits = c(0,1), expand = c(0,0)) +
  ggtitle("ROC Curve") +
  xlab("False Positive Rate") +
  ylab("True Positive Rate") +
  theme_classic()




############ bad model
# Build the random forest model
model <- randomForest(DAHAK10 ~ ., data = training_data, method = "class")

# Make predictions on the test data
predictions <- predict(model, test_data, type = "class")

# Evaluate the model's accuracy
accuracy <- mean(predictions == test_data$DAHAK10)
print(paste("Accuracy:", accuracy))
```

```
  geom_density(fill = "blue", color = "black") +
  xlab("Household Income") +
  ylab("Density") +
  ggtitle("Distribution of Household Income")

#NOT GOOD
#Plot the relationship between the type of house and household size using
a scatterplot:
ggplot(CleanData, aes(x = T.O, y = factor(M.O.B))) +
  geom_point(shape = 1) +
  xlab("Number of rooms in the house") +
  ylab("Type of house") +
  ggtitle("Relationship between Type of House and Household Size")

#Hexbin Plot: To visualize the distribution of two continuous variables
and show
#the density of points in a two-dimensional space, you can use a hexbin
plot.
hexbinplot(CleanData$sen ~ CleanData$M.INCOME, xlab = "Age", ylab = "Total
Income",
           main = "Hexbin Plot of Age vs Total Income",
           gridsize = 20, colramp = colorRampPalette(c("blue", "yellow",
"red")))

#Mosaic Plot: To show the relationship between two categorical variables,
you can use a mosaic plot.

mosaicplot(table(CleanData$jensiat, CleanData$DAHAK10), main = "Mosaic
Plot of Gender and Rich/Not Rich",
           color = TRUE, cex.axis = 0.6)


#Box-and-Whisker Plot with Jitter: To visualize the distribution of
multiple continuous variables for
#multiple categories, you can use a box-and-whisker plot with jitter.
ggplot(RawData, aes(x = as.factor(DAHAK10), y = M.INCOME, color =
as.factor(jensiat))) +
  geom_boxplot(outlier.shape = NA) +
  geom_jitter(width = 0.2, alpha = 0.5) +
  labs(x = "Rich/Not Rich", y = "Total Income", color = "Gender") +
  ggtitle("Box-and-Whisker Plot with Jitter of Total Income by Rich/Not
Rich and Gender")







#ArashSadeghiBablan


library(rpart)
library(lattice)
library(caret)
library(ROCR)
library(dplyr)
```