# گزارش پیاده‌سازی و مقایسه الگوریتم ID3

هیوا ابوالهادی زاده ۴۰۰۴۰۵۰۰۴
درس یادگیری ماشین

۲۵ آبان ۱۴۰۳

## مقدمه

هدف اصلی این تحلیل، ساخت و ارزیابی یک مدل درخت تصمیم برای پیش‌بینی متغیر هدف `salary` است. این متغیر به صورت دودویی تعریف شده است و دو مقدار ممکن دارد: `>50K` و `<=50K`، که به ترتیب نشان‌دهنده درآمد بالاتر یا کمتر از ۵۰ هزار دلار در سال است. برای این منظور از الگوریتم **ID3** استفاده شده است.

الگوریتم ID3 یک روش مشهور برای ساخت درخت تصمیم است که بر اساس آنتروپی و کسب اطلاعات **(Information Gain)** عمل می‌کند. این الگوریتم به طور خاص از داده‌های موجود برای شناسایی ویژگی‌هایی که بیشترین قدرت تفکیک را دارند، استفاده می‌کند و در نتیجه گره‌های درخت تصمیم را می‌سازد. هدف اصلی آن، تقسیم داده‌ها به گونه‌ای است که بالاترین خلوص ممکن در گره‌های نهایی (برچسب‌ها) حاصل شود.

داده‌های استفاده شده در این پروژه شامل ۱۳ ویژگی مختلف و ۳۲،۵۶۱ نمونه است که نمایانگر مشخصات افراد است. این ویژگی‌ها شامل اطلاعات جمعیت‌شناختی، شغلی و تحصیلی هستند.

این تحلیل در دو بخش اصلی انجام می‌شود:

۱. آماده‌سازی و پیش‌پردازش داده‌ها.

۲. ساخت و ارزیابی درخت تصمیم با استفاده از الگوریتم ID3.

این فرآیند به شناسایی عوامل کلیدی مؤثر بر درآمد افراد و همچنین ارزیابی توانایی مدل برای پیش‌بینی مقادیر هدف کمک می‌کند.

# شرح داده‌ها

این مجموعه داده شامل ۳۲،۵۶۱ نمونه و ۱۳ ویژگی است که ترکیبی از داده‌های پیوسته و گسسته را شامل می‌شود. هدف اصلی، پیش‌بینی متغیر هدف salary است که درآمد افراد را در دو سطح 50K< و 50K=> دسته‌بندی می کند.

## تجزیه‌وتحلیل ستون‌ها

- ویژگی‌های پیوسته: ستون‌هایی مانند age، fnlwgt، capital-gain، capital-loss، و hours-per-week تنوع بالایی از مقادیر را نشان می‌دهند و نشان‌دهنده متغیرهای عددی با دامنه‌های گسترده هستند. - ویژگی‌های گسسته: ستون‌هایی مانند workclass، education، marital-status، occupation، relationship، race، sex، native-country، و salary دارای تعداد محدودی از مقادیر هستند و نشان‌دهنده دسته‌بندی‌های خاص هستند.

## حذف ویژگی‌های غیرضروری

ویژگی‌های fnlwgt و education-num به دلیل عدم تأثیرگذاری مستقیم بر مدل و ارتباط بالا با سایر ویژگی‌ها حذف شده‌اند. education-num اطلاعات تکراری از ستون education ارائه می‌دهد.

## مدیریت داده‌های گم‌شده

در ستون‌هایی که مقادیر "?" به‌عنوان مقادیر گم‌شده نشان داده شده‌اند (مانند workclass و occupation)، از مقدار mod هر ستون برای پر کردن این داده‌ها استفاده شده است. این روش باعث کاهش سوگیری و حفظ انسجام داده‌ها شده است.

## دسته‌بندی داده‌ها

تمام ویژگی‌ها به مقادیر گسسته دسته‌بندی شده‌اند تا سازگاری بیشتری با الگوریتم درخت تصمیم فراهم شود. این فرآیند شامل تقسیم ویژگی‌های پیوسته به بازه‌های مشخص و تخصیص مقادیر متناسب بوده است.

# توزیع ویژگی‌ها

براساس شکل ۱، توزیع ویژگی‌های مختلف در داده‌ها به شکل زیر قابل توضیح است:

۱. ویژگی تحصیلات (**Education**):

- بیشتر افراد در دسته‌های تحصیلی مانند "HS-grad" و "Some-college" قرار دارند، که نشان می‌دهد سطح تحصیلات عمده افراد در سطح متوسط قرار دارد.

دسته‌های تحصیلی مانند "Doctorate" و "Preschool" تعداد کمی از افراد را شامل می‌شوند.

۲. ویژگی نژاد (**Race**):

- بیشتر افراد در این مجموعه داده سفیدپوست هستند. سایر نژادها مانند "Black" و "Asian-Pac-Islander" در تعداد کمتری حضور دارند، و دسته "Other" و "Amer-Indian-Eskimo" کمترین تعداد افراد را شامل می‌شوند.

۳. ویژگی وضعیت ازدواج (**Marital Status**):

- بیشتر افراد در دسته‌های "Married-civ-spouse" و "Never-married" قرار دارند که نشان‌دهنده تعداد بالای افراد متأهل و هرگز ازدواج نکرده است. دسته‌های "Divorced" و "Widowed" در تعداد کمتری حضور دارند.

۴. ویژگی جنسیت (**Sex**):

- تعداد مردان در مجموعه داده بسیار بیشتر از زنان است. این ویژگی نشان‌دهنده عدم تساوی جنسیتی در داده‌ها است.

۵. ویژگی ساعت کاری در هفته (**Hours per Week**):

- بیشتر افراد به صورت تمام‌وقت (Full-time) کار می‌کنند، و تعداد کمتری به صورت پاره‌وقت (Part-time) و اضافه‌کاری (Overtime) مشغول به کار هستند.

۶. ویژگی درآمد از سرمایه (**Capital Gain**):

- اکثر افراد هیچ درآمدی از سرمایه ندارند (No Gain)، و تعداد کمی از افراد در دسته‌های "Low Gain"، "Medium Gain"، و "High Gain" قرار دارند.

۷. ویژگی زیان سرمایه (**Capital Loss**):

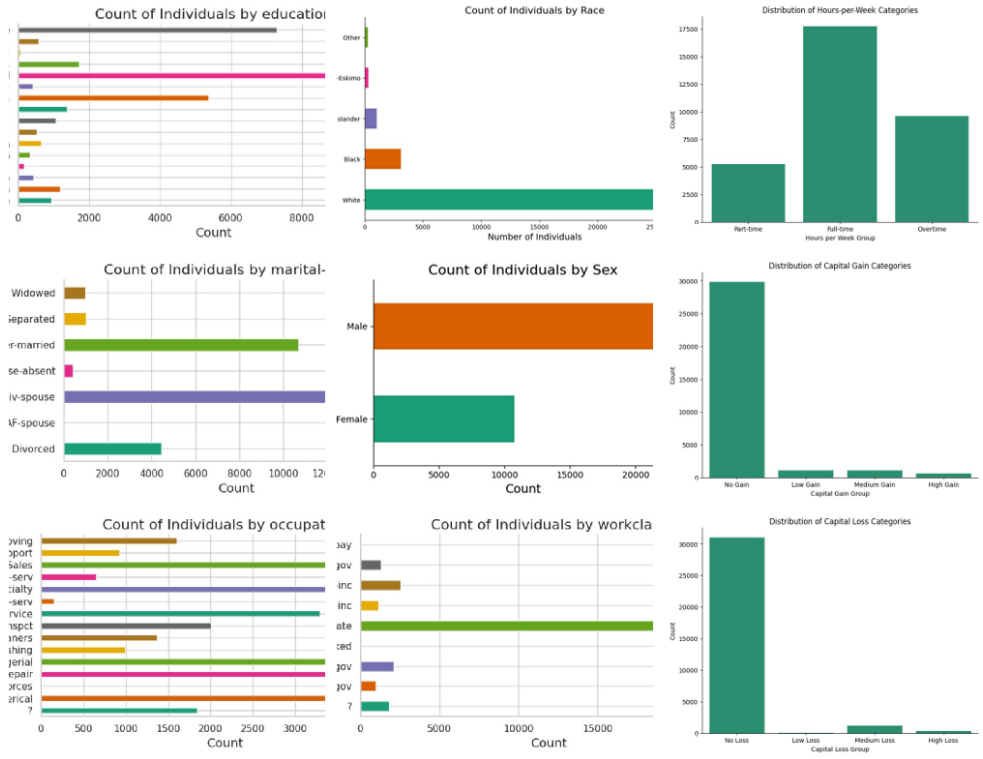
- مانند ویژگی درآمد از سرمایه، اکثر افراد هیچ زیانی از سرمایه ندارند (No Loss)، و تعداد کمی از افراد دارای زیان در دسته‌های "Low Loss"، "Medium Loss"، و "High Loss" هستند.

۸. ویژگی شغل (**Occupation**):

- بیشترین تعداد افراد در دسته‌های شغلی "Prof-specialty" و "Exec-managerial" قرار دارند، که نشان‌دهنده حضور بالای افراد در موقعیت‌های شغلی حرفه‌ای و مدیریتی است. دسته‌هایی مانند "Priv-house-serv" و "Armed-Forces" تعداد بسیار کمی از افراد را شامل می‌شوند.

۹. ویژگی نوع شغل (**Workclass**):

- بیشتر افراد در دسته ”Private“ قرار دارند که بیانگر شغل در بخش خصوصی است. سایر دسته‌ها مانند ”Self-emp-not-inc“ و ”Local-gov“ در تعداد کمتری وجود دارند.




شکل ۱: نمودارهای توزیع ویژگی‌ها

در شکل ۲، سه ویژگی رابطه خانوادگی (Relationship)، حقوق (Salary)، و سن (Age) توزیع شده‌اند:

۱. ویژگی رابطه خانوادگی (**Relationship**):

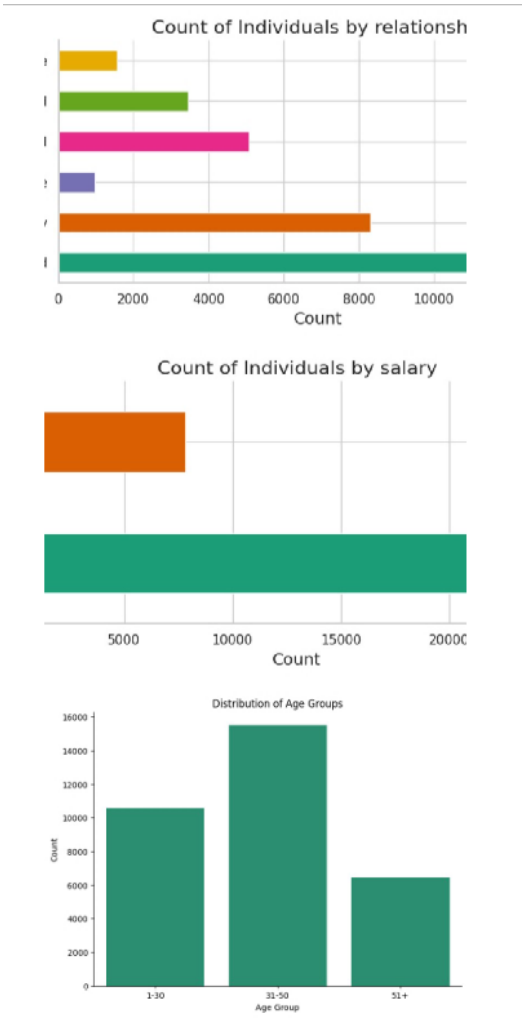
- بیشترین تعداد افراد در دسته‌های ”Husband“ و ”Not-in-family“ قرار دارند. این نشان می‌دهد که بسیاری از افراد یا در نقش همسر هستند یا ارتباط خانوادگی مستقیمی ندارند. دسته‌های ”Unmarried“ و ”Wife“ در تعداد کمتری حضور دارند.

۲. ویژگی حقوق (**Salary**):

- تعداد بیشتری از افراد درآمد سالانه کمتر از ۵۰ هزار دلار دارند (<=50K)، در حالی که تعداد کمتری درآمد بالای ۵۰ هزار دلار دارند (>50K). این نشان می‌دهد که توزیع درآمد به سمت درآمدهای کمتر تمایل دارد.

۳. ویژگی سن (**Age**):

- بیشتر افراد در گروه سنی ۳۱ تا ۵۰ سال قرار دارند که نشان‌دهنده‌ی گروه سنی فعال و غالب در داده‌هاست. پس از آن، گروه سنی ۱ تا ۳۰ سال قرار دارد و افراد بالای ۵۰ سال کمترین تعداد را دارند.




شکل ۲: نمودارهای توزیع ویژگی‌ها

# ساخت و پیاده‌سازی درخت تصمیم

در این بخش به شرح مراحل ساخت و پیاده‌سازی درخت تصمیم می‌پردازیم. درخت تصمیم به صورت ساختار داده‌ای دیکشنری پیاده‌سازی شده است، که در آن هر گره به عنوان یک کلید در دیکشنری تعریف شده و مقادیر آن گره‌های فرزند را مشخص می‌کنند.

برای ایجاد این مدل، الگوریتم ID3 انتخاب شده است. این الگوریتم با استفاده از معیار Information Gain به انتخاب ویژگی‌های مناسب برای تقسیم داده‌ها می‌پردازد. در هر گام، ویژگی‌ای که بیشترین اطلاعات را ارائه می‌دهد، به عنوان گره تصمیم انتخاب می‌شود. این فرایند به صورت بازگشتی ادامه پیدا می‌کند تا زمانی که به شرایط توقف (مانند رسیدن به کلاس یکتا یا محدودیت عمق درخت) برسد.

## دلایل انتخاب **ID3**

انتخاب این الگوریتم به دلیل سادگی و کارایی آن در داده‌های گسسته صورت گرفته است. ID3 به طور موثری می‌تواند داده‌ها را براساس ویژگی‌های توصیفی دسته‌بندی کند و ساختار ساده‌تری نسبت به الگوریتم‌های پیچیده‌تر ارائه می‌دهد.

## چگونگی استفاده از مدل

پس از ساخت درخت، مدل ایجاد شده را می‌توان برای پیش‌بینی کلاس‌های جدید استفاده کرد. این کار با حرکت در طول شاخه‌های درخت و بررسی مقادیر ویژگی‌ها برای نمونه جدید انجام می‌شود تا به یک گره برگ که نشان‌دهنده کلاس پیش‌بینی شده است، برسیم.

# **Encode** کردن داده‌ها

در مدل‌های یادگیری ماشین، داده‌ها معمولاً شامل ویژگی‌های عددی و دسته‌بندی (طبقه‌بندی) هستند. از آنجایی که اکثر الگوریتم‌های یادگیری ماشین تنها با داده‌های عددی سازگار هستند، ویژگی‌های دسته‌بندی باید به نحوی به شکل عددی تبدیل شوند. این فرآیند به عنوان Encode کردن شناخته می‌شود و هدف آن تبدیل داده‌های غیرعددی به مقادیر عددی قابل‌پردازش برای مدل است.

در این پروژه، از سه روش اصلی برای Encode کردن داده‌های دسته‌ای استفاده شده است:

۱. One-Hot Encoding: این روش هر مقدار دسته‌ای را به یک ستون جدید تبدیل کرده و با مقدار ۱ یا ۰ پر می‌کند. این روش زمانی مناسب است که ویژگی موردنظر دارای چندین دسته‌بندی است و ترتیب بین دسته‌ها اهمیتی ندارد.

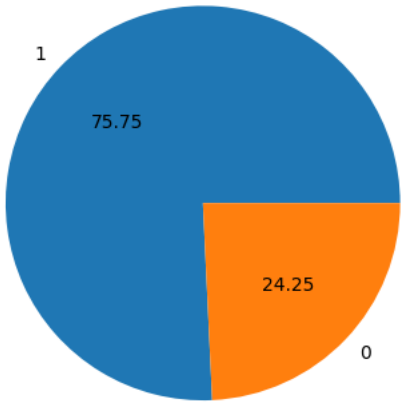
۲. Label Encoding: در این روش، هر دسته یک عدد منحصر به فرد اختصاص داده می‌شود. این روش برای ویژگی‌های با مقادیر متعددی که دارای ترتیب یا رتبه‌بندی نیستند، مناسب است.

۳. Ordinal Encoding: این روش زمانی استفاده می‌شود که دسته‌ها دارای نوعی ترتیب یا سلسله‌مراتب باشند. هر دسته با یک عدد خاص مرتبط شده که نشان‌دهنده جایگاه یا رتبه آن است.

این روش‌ها به مدل کمک می کنند تا داده‌های دسته‌ای را به‌درستی پردازش کرده و از آن‌ها در یادگیری استفاده کند.

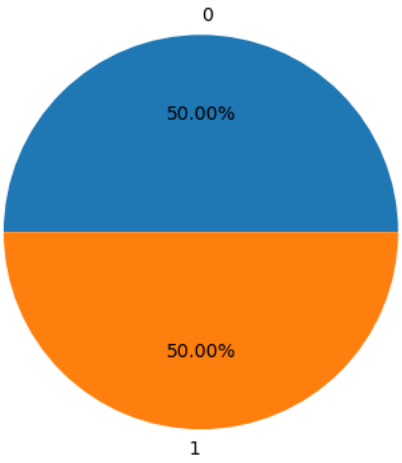
## عدم توازن (Imbalance) در داده‌ها و اهمیت بالانس کردن آن

در بسیاری از مسائل یادگیری ماشین، داده‌ها به صورت نامتوازن هستند. به این معنی که تعداد نمونه‌های یک کلاس (یا چند کلاس) بیشتر از سایر کلاس‌ها است. در شکل ۳ میتوانید نسبت کلاس های دیتا را مشاهده کنید.




شکل ۳: نسبت کلاس‌ها

برای مقابله با این مشکل، روش‌های مختلفی وجود دارد که یکی از آن‌ها Upsampling است. در این روش، نمونه‌های کلاس با تعداد کمتر به طور مصنوعی چند برابر می‌شوند تا تعداد نمونه‌ها در هر کلاس به یک سطح متوازن نزدیک‌تر شود. این کار باعث می‌شود که مدل بتواند همه‌ی کلاس‌ها را بهتر یاد بگیرد و عملکرد کلی آن بهبود یابد.




شکل ۴: نسبت کلاس‌ها بعد از متعادل کردن

## بررسی همبستگی ویژگی‌ها با متغیر هدف

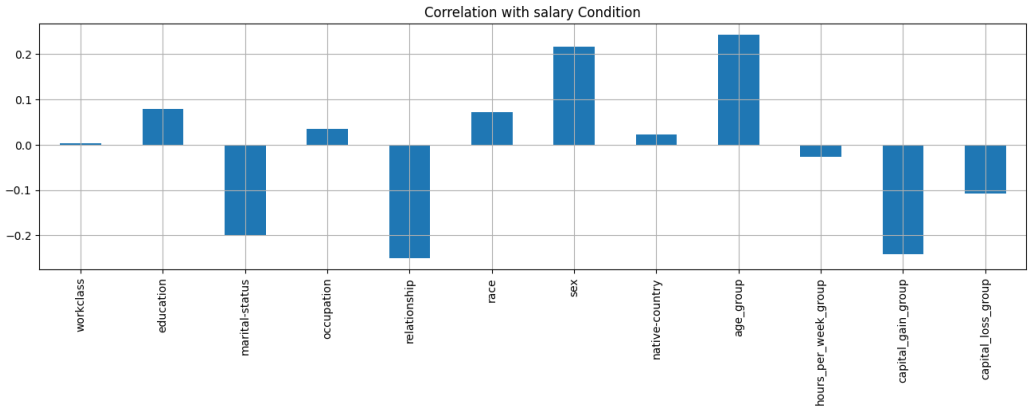
در تحلیل داده‌های خود، به منظور ارزیابی اهمیت و ارتباط ویژگی‌های مختلف با متغیر هدف (که در اینجا متغیر ”salary” است)، از ضریب همبستگی (”correlation”) استفاده کردیم. این تحلیل به ما کمک می‌کند تا میزان رابطه و تاثیرگذاری هر ویژگی را نسبت به متغیر هدف مشخص کنیم. در شکل ۵، همبستگی هر ویژگی با ”salary” نمایش داده شده است.

ویژگی‌هایی با همبستگی مثبت مانند ”sex” و ”age_group” نشان می‌دهند که این عوامل ممکن است بر حقوق تاثیر مثبتی داشته باشند؛ یعنی با افزایش مقادیر این ویژگی‌ها، احتمال افزایش حقوق نیز وجود دارد. از سوی دیگر، ویژگی‌هایی مانند ”marital-status”، ”relationship” و ”capital_loss_group” دارای همبستگی منفی هستند و ممکن است تأثیر معکوس بر روی حقوق داشته باشند.

برای بررسی دقیق‌تر و ارزیابی تاثیر این ویژگی‌ها بر روی عملکرد مدل، یک زیرمجموعه (”subset”) از ویژگی‌های انتخاب شده را ایجاد کردیم که شامل ویژگی‌های زیر است:

”relationship”, ”marital-status”, ”sex”, ”age_group”, ”capital_gain_group”

این انتخاب به ما کمک می‌کند تا با تمرکز بر روی ویژگی‌های مهم‌تر، دقت و کارایی مدل را ارزیابی و بهینه کنیم.




شکل ۵: همبستگی داده ها

# ۱ نتایج و ارزیابی مدل

در این بخش، الگوریتم را با استفاده از شش روش مختلف روی داده‌ها تست کردیم. هر روش با جزئیات نتایج ارزیابی آن توصیف شده است.

- روش اول: one hot encoding روی داده‌ها اعمال شد. تعداد ویژگی‌ها به 102 افزایش یافت که باعث شد مدل اصلاً قابل اجرا نباشد.
- روش دوم: داده‌ها با استفاده از label encoding کدگذاری شدند. ارتفاع درخت برابر با 24 بود و نتایج به شرح زیر است:
  - Accuracy: 0.83
  - Confusion Matrix:

$$\begin{bmatrix} 6363 & 590 \\ 926 & 1163 \end{bmatrix}$$

  - Precision: 0.66
  - Recall: 0.56
  - F1-Score: 0.61
- روش سوم: داده‌ها با استفاده از ordinal encoding کدگذاری شدند. ارتفاع درخت برابر با 24 بود و نتایج به شرح زیر است:
  - Accuracy: 0.837
  - Confusion Matrix:

$$\begin{bmatrix} 6363 & 590 \\ 926 & 1163 \end{bmatrix}$$

- Precision: 0.66
- Recall: 0.56
- F1-Score: 0.61

- روش چهارم: زیرمجموعه‌ای از ویژگی‌ها با همبستگی بالا انتخاب شد. ارتفاع درخت برابر با 10 بود و نتایج به شرح زیر است:
  - Accuracy: 0.81
  - Confusion Matrix:

$$\begin{bmatrix} 7285 & 164 \\ 1737 & 572 \end{bmatrix}$$

  - Precision: 0.78
  - Recall: 0.25
  - F1-Score: 0.38

- روش پنجم: از یک زیرمجموعه تصادفی از ویژگی‌ها استفاده شد. ارتفاع درخت برابر با 10 بود و نتایج به شرح زیر است:
  - Accuracy: 0.795
  - Confusion Matrix:

$$\begin{bmatrix} 6697 & 734 \\ 1255 & 1053 \end{bmatrix}$$

  - Precision: 0.589
  - Recall: 0.456
  - F1-Score: 0.514

- روش ششم: داده‌ها با استفاده از upsampling متوازن شدند. ارتفاع درخت برابر با 24 بود و نتایج به شرح زیر است:
  - Accuracy: 0.87
  - Confusion Matrix:

$$\begin{bmatrix} 5592 & 1223 \\ 595 & 6813 \end{bmatrix}$$

  - Precision: 0.847
  - Recall: 0.919
  - F1-Score: 0.8822

# جدول مقایسه نتایج

در جدول‌های زیر، نتایج معیارهای ارزیابی برای هر روش به صورت جداگانه نمایش داده شده است.

جدول ۱: دقت (Accuracy)

| Accuracy | روش |
|---|---|
| 0.8323 | Label Encoding |
| 0.8323 | Ordinal Encoding |
| 0.8052 | High Correlation Features |
| 0.7958 | Random Features Subset |
| 0.8722 | Balanced Data |

جدول ۲: دقت (Precision)

| Precision | روش |
|---|---|
| 0.6634 | Label Encoding |
| 0.6634 | Ordinal Encoding |
| 0.7772 | High Correlation Features |
| 0.5893 | Random Features Subset |
| 0.8478 | Balanced Data |

جدول ۳: یادآوری (Recall)

| Recall | روش |
|---|---|
| 0.5567 | Label Encoding |
| 0.5567 | Ordinal Encoding |
| 0.2477 | High Correlation Features |
| 0.4562 | Random Features Subset |
| 0.9197 | Balanced Data |

جدول ۴: نمره F1-Score

| روش | F1-Score |
|---|---|
| Label Encoding | 0.6054 |
| Ordinal Encoding | 0.6054 |
| High Correlation Features | 0.3757 |
| Random Features Subset | 0.5143 |
| Balanced Data | 0.8823 |

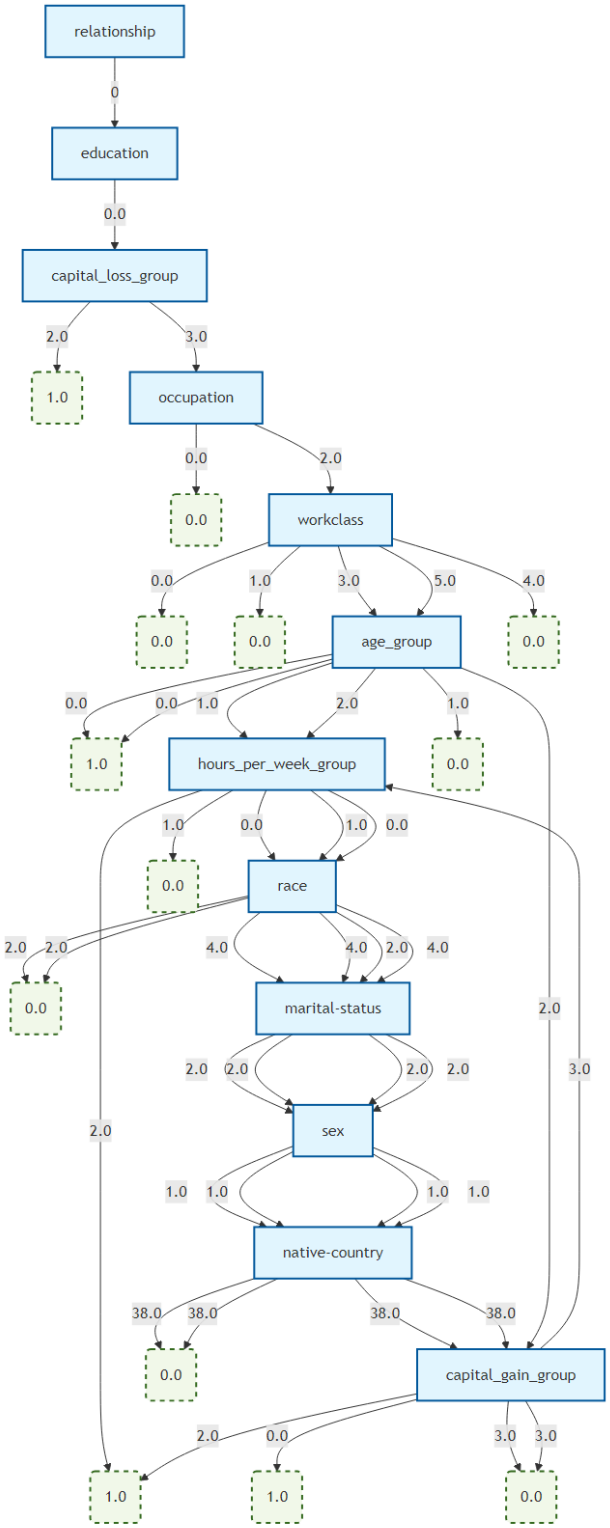
## ۲ نتیجه‌گیری

در این گزارش، الگوریتم ID3 را بر روی یک مجموعه داده با روش‌های مختلف پیش‌پردازش و تعدیل اجرا کردیم و معیارهای ارزیابی مختلفی از جمله دقت (Accuracy)، دقت مثبت (Precision)، یادآوری (Recall)، و نمره F1-Score را برای هر روش محاسبه کردیم. نتایج حاصل به شرح زیر است:

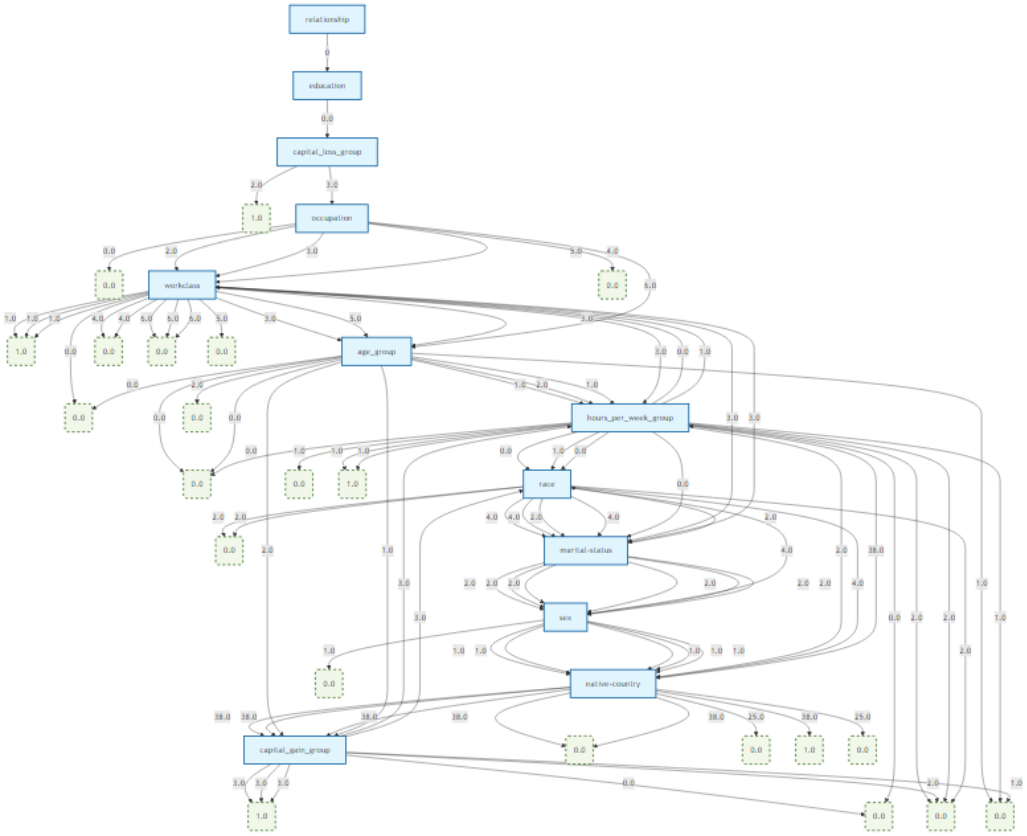
- **Label Encoding** و **Ordinal Encoding**: این دو روش مشابه بوده و نتایج یکسانی به دست داده‌اند. هر دو روش دارای دقتی برابر با 0.8323 بودند و نمره F1-Score آن‌ها برابر با 0.6054 بوده است. اگرچه این روش‌ها نتایج متوسطی ارائه داده‌اند، اما میزان یادآوری پایینی نسبت به دیگر روش‌ها داشتند (0.5567).
- استفاده از فیچرهای با همبستگی بالا: در این روش، تنها زیرمجموعه‌ای از فیچرهای مرتبط با خروجی مدل انتخاب شد. دقت (Accuracy) در این روش به 0.8052 کاهش یافت و یادآوری (Recall) نیز به 0.2477 محدود شد که نشان‌دهنده کاهش توان مدل در شناسایی نمونه‌های مثبت است. اگرچه دقت مثبت (Precision) افزایش یافته و به 0.7772 رسیده است، اما نمره F1-Score نسبتاً پایین‌تر و برابر با 0.3757 بوده که نشان‌دهنده عدم تعادل بین دقت و یادآوری است.
- زیرمجموعه تصادفی از فیچرها: این روش نیز مانند روش قبلی با کاهش دقت مواجه شد (0.7958) و نمره F1-Score آن برابر با 0.5143 بود. این روش منجر به کاهش دقت مثبت (Precision) و یادآوری (Recall) شد که نشان‌دهنده کاهش توانایی مدل در شناسایی و دسته‌بندی درست نمونه‌ها است.
- دیتای بالانس شده: این روش بهترین نتایج را ارائه داده است. با دقت (Accuracy) برابر با 0.8722، این روش بهترین عملکرد را در میان روش‌های مختلف داشته است. همچنین، نمره F1-Score برابر با 0.8823 و یادآوری (Recall) برابر با 0.9197 نشان می‌دهد که این روش تعادل بهتری بین دقت و یادآوری ایجاد کرده و به مدل کمک کرده تا نمونه‌های مثبت را به طور موثری شناسایی کند.

نتیجه‌گیری کلی: استفاده از داده‌های بالانس شده بهترین نتیجه را در تمامی معیارها داشته است. این نشان می‌دهد که بالانس‌سازی داده‌ها در بهبود عملکرد الگوریتم ID3 تاثیر به‌سزایی دارد

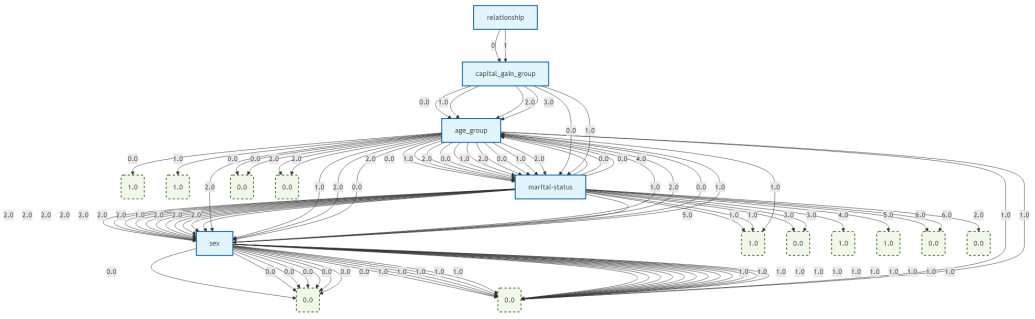
و می‌تواند به کاهش نرخ خطا و بهبود تعادل بین دقت و یادآوری کمک کند. به طور کلی، می‌توان نتیجه گرفت که بالانس کردن داده‌ها رویکردی مناسب برای افزایش دقت و یادآوری در مسائلی است که داده‌های نامتوازن دارند.




شکل ۶: بخشی از درخت روش دوم




شکل ۷: بخشی از درخت روش سوم




شکل ۸: بخشی از درخت روش چهارم