ПОСТАПКА

Saturday, February 1, 2025 9:43 PM

Пример за полесно разбирање:
Замислете датасет за предвидување на платата на вработените врз основа на:
Карактеристики (features):
Возраст (нумеричка)
Искуство (нумеричка)
Пол (категориска: Машко/Женско)
Позиција (категориска: Јуниор/Сениор/Менаџер)
Пример за target променливи:

- Категориска: Бинарна (дали ќе добие покачување на плата, унапредување ДА/НЕ) или Мултикласна (ниска/средна/висока плата)
- Нумеричка: Плата (континуирана, на пр., 50000 $)

Пример со бинарна таргет променлива:
Кога се справуваме со предвидување на бинарна категориска целна променлива (на пр., дали вработениот ќе добие промоција Да/Не), прво треба да одлучиме со кои карактеристики(features) треба да работиме а тоа зависи од неколку фактори:

1. Анализа на карактеристиките
а) Важност на карактеристиките (Feature Relevance):
Не сите карактеристики (како возраст, искуство, пол) можат да бидат релевантни за одлуката.
Пример: Искуство и Возраст веројатно имаат директно влијание врз одлуката за промоција, додека пол можеби нема значајно влијание.

- Кај некои дата сетови можеме и на око да процениме дали е важна некоја карактеристика па ако стварно нема никакво влијание кон таргетот ја отстрануваме уште од почеток, но во покомплицирани дата сетови не е очигледно уште од почеток дали треба да се отстрани или не па затоа разгледуваме неколку работи:

б) Корелација со таргетот:
Мериме корелација меѓу секоја карактеристика и таргетот (Да/Не).
*** Битно е да имаме унифициран пристап при мерење корелација, затоа ако има мешани карактеристики како во примерот прво ги енкодираме да бидат сите нумерички па користиме метрики за корелација (пр. Correlation matrix)
+ ИНФО ЗА ЕНКОДИРАЊЕ:

- Изборот на енкодер е многу важен
  Ако користиме Label encoding за пол карактеристиката може да воведе лажен редослед, ќе додели пр. Машко=0, женско=1 а реално тие немаат природен редослед
  Затоа во случаи каде карактеристиката нема природен редослед, се користи Target (за мал број на категории како во случајов машко/женско) или One-Hot Encoding (за многу категории)

Анализирај ги корелационите вредности:

- Висока позитивна или негативна вредност укажува на силна врска.
- Вредности блиску до 0 укажуваат на слаба или никаква врска.

Потенцијални проблеми со корелација:

- Корелацијата (особено Pearson) мери само линеарни односи. Ако има сложени нелинеарни односи, корелацијата можеби нема да биде корисна.
- Решение: Користи Decision Trees за автоматски избор на важни карактеристики.

- Друг проблем е **превисока корелација,** не мора да значи дека висока корелација = зависност = ни треба таа колона туку баш напротив, ако имаат корелација многу блиску до 1 тогаш најверојатно се работи за исти податоци и можеме да отфрлиме една од нив.

в) Карактеристики со мала варијанса:
Ако карактеристиката има речиси иста вредност за сите примероци, таа е нерелевантна и треба да се отстрани.

2. Процес на избор на карактеристики
Прво треба да процениме дали ќе ги користиме сите карактеристики или само дел од нив

1) **Сите карактеристики:**
   Ги користиме сите ако не сме сигурни кои се релевантни и ако користиме модел како Random Forest, кој сам одбира релевантни карактеристики.
2) **Само релевантни:**
   Ако користиме модели како Logistic Regression, треба претходно да отстраниме нерелевантни карактеристики, бидејќи тие можат да ја намалат точноста.
   Задржи само важни карактеристики --> процесот со проверка на корелација
   - Feature selection
   - Drop на колоните што ќе пресудиме дека се нерелевантни

Анализа на резултатите
Корелациони вредности:

Највисоките позитивни или негативни корелации укажуваат на карактеристики што најмногу влијаат на таргетот.
На пр., ако Age има висока позитивна корелација, тоа значи дека возраст е силно поврзана со веројатноста за промоција.
Карактеристики со мала корелација:

Ако карактеристиката има корелација блиску до 0, можеш да размислиш за нејзино отстранување.

- Ако спефицично се бара во задача да се користи Decision Tree модел тогаш нема потреба од проверка на корелација, feature selection како што би правеле со линеарна регресија

Различни модели бараат различен пристап кога станува збор за проверка на корелација, избор на карактеристики (Feature Selection) и специфицирање на дополнителни метрики.

1️⃣ Линеарни модели (Регресија и Класификација)

Модели:

- Linear Regression
- Logistic Regression
- Lasso Regression / Ridge Regression
- Support Vector Machines (SVM) (ако се користи линеарно јадро)

Прашање	Одговор
Треба ли проверка на корелација?	✅ **Да** – Линеарните модели се **многу осетливи** на мултиколинеарност (ако две или повеќе карактеристики се силно корелирани).
Треба ли Feature Selection?	✅ **Да** – Линеарните модели немаат вграден механизам за избор на најважните карактеристики. Најчесто се користат **Lasso (L1 регуларизација)** или **SelectKBest** за избор на најдобрите карактеристики.
Треба ли спецификација на дополнителна метрика?	✅ **Да**, особено за **Logistic Regression**, каде што Accuracy не е доволна. Обично се користат **Precision**, **Recall**, **F1-score** или **ROC-AUC**.

🔹 **Пример за корелација пред линеарен модел:**

python Copy Edit

```python
import seaborn as sns
import matplotlib.pyplot as plt

# Приказ на корелациона матрица
plt.figure(figsize=(8,6))
sns.heatmap(df.corr(), annot=True, cmap="coolwarm")
plt.show()
```

2️⃣ Decision Trees и Random Forests

Модели:

- Decision Tree Classifier / Regressor
- Random Forest
- Extra Trees
- Gradient Boosting (XGBoost, LightGBM, CatBoost)

Прашање	Одговор
Треба ли проверка на корелација?	❌ **Не** – Decision Trees и Random Forests **не се осетливи** на мултиколинеарност.
Треба ли Feature Selection?	🔹 **Зависи** – Decision Trees сами одбираат важни карактеристики, но ако имаш **многу карактеристики (100+), можеби треба да отстраниш неважни** за подобра ефикасност.
Треба ли спецификација на дополнителна метрика?	✅ **Да**, особено за Random Forest и Gradient Boosting, каде што **Accuracy не е секогаш доволна**. Често се користат **F1-score**, **Precision**, **Recall** за класификација и **RMSE или R^2** за регресија.

🔹 **Пример за добивање на важноста на карактеристиките пред да се отстранат неважните:**

python Copy Edit

```python
from sklearn.ensemble import RandomForestClassifier
import pandas as pd

# Тренирање на модел
rf = RandomForestClassifier()
rf.fit(X_train, y_train)

# Приказ на важноста на карактеристиките
feature_importance = pd.Series(rf.feature_importances_, index=X_train.columns)
print(feature_importance.sort_values(ascending=False))
```

👉 Ако некоја карактеристика има **многу ниска важност**, можеш да ја отстраниш.

3️⃣ Нелинеарни модели (SVM, Neural Networks)

Модели:

- **Support Vector Machines (SVM)** (со нелинеарни јадра, `rbf`, `poly`)
- **Neural Networks (MLP, TensorFlow, PyTorch)**

Прашање	Одговор
Треба ли проверка на корелација?	✅ **Да** за **SVM** – Може да биде проблем ако карактеристиките се силно корелирани, но **не е неопходно за Neural Networks**.
Треба ли Feature Selection?	❌ **Не** за **Neural Networks** – Невронските мрежи сами учат најдобри карактеристики, но **може да помогне за SVM** ако има многу карактеристики.
Треба ли спецификација на дополнителна метрика?	✅ **Да**, особено за **Neural Networks**, каде што се користат **Accuracy, Precision, Recall, F1-score или ROC-AUC за класификација** и **MSE/RMSE за регресија**.

🔷 **Пример за спецификација на метрика во Neural Network (TensorFlow/Keras):**

python Copy Edit

```python
import tensorflow as tf
from tensorflow import keras

# Дефинирање на моделот
model = keras.Sequential([
    keras.layers.Dense(64, activation="relu"),
    keras.layers.Dense(32, activation="relu"),
    keras.layers.Dense(1, activation="sigmoid")  # За бинарна класификација
])

# Компилација со дополнителни метрики (F1-score се имплементира посебно)
model.compile(optimizer="adam", loss="binary_crossentropy", metrics=["accuracy", tf.keras.
```

ФИНАЛНО:

Модел	Корелација?	Feature Selection?	Дополнителна метрика?
Linear Regression	✅ Да	✅ Да	✅ R^2, RMSE
Logistic Regression	✅ Да	✅ Да	✅ Precision, Recall, F1-score
Lasso/Ridge Regression	✅ Да	❌ Не (Lasso автоматски прави селекција)	✅ R^2, RMSE
Decision Tree	❌ Не	🔷 Зависи (ако има 100+ карактеристики)	✅ Precision, Recall, F1-score
Random Forest	❌ Не	🔷 Зависи (Feature Importance Analysis)	✅ Precision, Recall, F1-score
XGBoost, LightGBM, CatBoost	❌ Не	❌ Не (самите модели одбираат важни карактеристики)	✅ Precision, Recall, F1-score
SVM (Linear Kernel)	✅ Да	✅ Да	✅ Precision, Recall, F1-score
Neural Networks (MLP, TensorFlow, PyTorch)	❌ Не	❌ Не	✅ Precision, Recall, F1-score