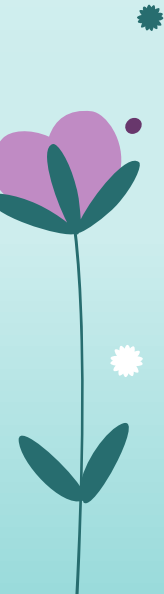


Regression Models Performance Evaluation

&

Feature Importance & confusion matrix

يعرض هذا القسم نتائج تقييم نماذج الانحدار المستخدمة و مقارنتها من حيث الاداء



KNN Regression Model

```
● print("MAE:", mae)  
  print("RMSE:", rmse)  
  print("R2:", r2)
```

[01]

..

```
MAE: 2.356940507888794  
RMSE: 15.84200668334961  
R2: 0.8552253842353821
```

قمنا بتقييم اداء نموذج KNN Regression Model ب استخدام مقاييس R^2 & MAE & RMSE. أظهرت النتائج أن قيمة R^2 بلغت 0.85، مما يدل على أن النموذج يفسر نسبة كبيرة من التغير في البيانات.

كما كانت قيمة MAE منخفضة، مما يعني أن متوسط الخطأ بين القيم الحقيقية والمتوقعة صغير.

بناءً على هذه النتائج تم اعتماد هذا النموذج ك نموذج نهائي للأنحدار):

Decision Tree Regression Model

```
model_score(final_DTR_pip , xr_train , yr_train , xr_test , yr_test)
```

```
Warning: No categorical columns found. Calling 'transform' will only return input data.
```

```
MAE: 3.56681374869652
```

```
RMSE: 5.655502131743462
```

```
R2: 0.7077028075321873
```

استخدمنا نموذج **Decision Tree Regression** بعد ما عدّلنا الباراميترز تبعته عشان نطلع بأفضل نتيجة. الهدف كان نقلل الخطأ ونحسن التوقعات.

من النتائج بنلاحظ إن قيمة R^2 حوالي **0.71**، وهذا يعني إن النموذج بفهم جزء كبير من البيانات وبعطي توقعات منيحة. كمان قيم **MAE** و **RMSE** أقل من بعض النماذج الثانية، وهذا يدل إن الأخطاء بالتوقع مش كبيرة.

بشكل عام، نموذج **Decision Tree** أدائه جيد وقدر يتعامل مع البيانات بشكل أفضل، خصوصًا بالعلاقات غير الخطية

Random Forest Regression Model

```
R2: 0.7189143259989955
```

```
model_score(final_RFR_pipeline , xr_train , yr_train , xr_test , yr_test) |  
r2_train = final_RFR_pipeline.score(xr_train, yr_train)
```

164]

```
... MAE: 3.563659235214629  
RMSE: 5.546379178723631  
R2: 0.7188737388072082
```

```
... 'MAE: 3.561494489024377\nRMSE: 5.545978789090631\nR2: 0.7189143259989955'
```

استخدمنا نموذج Random Forest عشان أتوقع القيم، وبالنهاية طلعت معي هاي النتائج:

❖ MAE حوالي 3.5: يعني بشكل عام التوقعات بتغلط تقريبًا 3 أو 4 وحدات عن القيمة الحقيقية، وهذا إشي مقبول.

❖ RMSE حوالي 5.5: هذا بيوضح إن في شوية أخطاء كبيرة، بس بشكل عام الأداء لسه منيح.

❖ R^2 حوالي 0.71: هذا أهم رقم، وببديل إن النموذج فاهم تقريبًا 71% من البيانات، يعني شغله كويس.

نموذج Random Forest أدائه منيح، وأحسن من بعض النماذج الثانية، وبنقدر نعتمد عليه بالتوقعات

XGBoost Regression Model

```
print("MAE:", mean_absolute_error(yr_test, y_pred_test))  
print("RMSE:", mean_squared_error(yr_test, y_pred_test))  
print("R2:", r2_score(yr_test, y_pred_test))
```

[176]

...

MAE: 3.783957004547119

RMSE: 30.230783462524414

R2: 0.723731279373169

اعتمدنا على MAE و R^2 لتقييم النموذج.

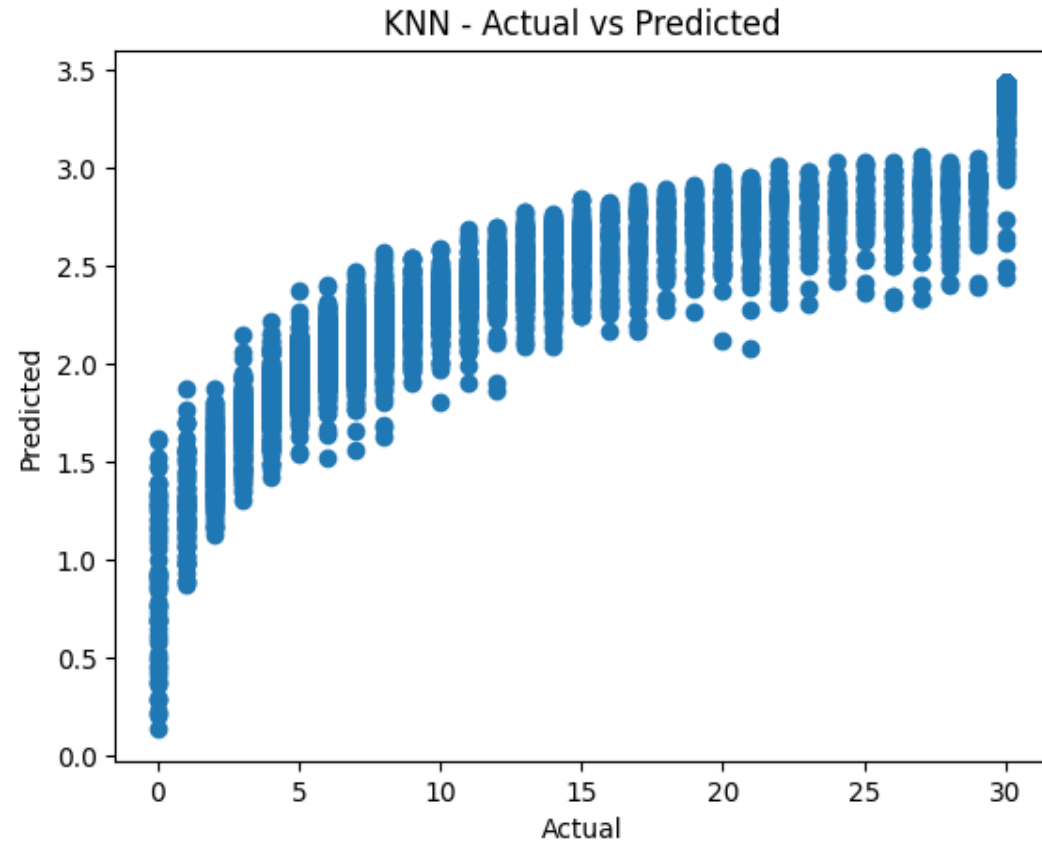
قيمة MAE تقريبا 3.7: بمنخفضة نسبياً، وهذا يدل إن التوقعات قريبة من القيم الحقيقية.

كما أن قيمة R^2 تقريبا 0.72: جيدة، مما يعني **أن النموذج** قادر على تفسير جزء كبير من البيانات

Regression Model Evaluation And Explainability

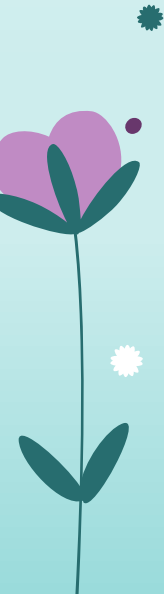
Feature Importance & confusion matrix

KNN – Actual VS Predicted

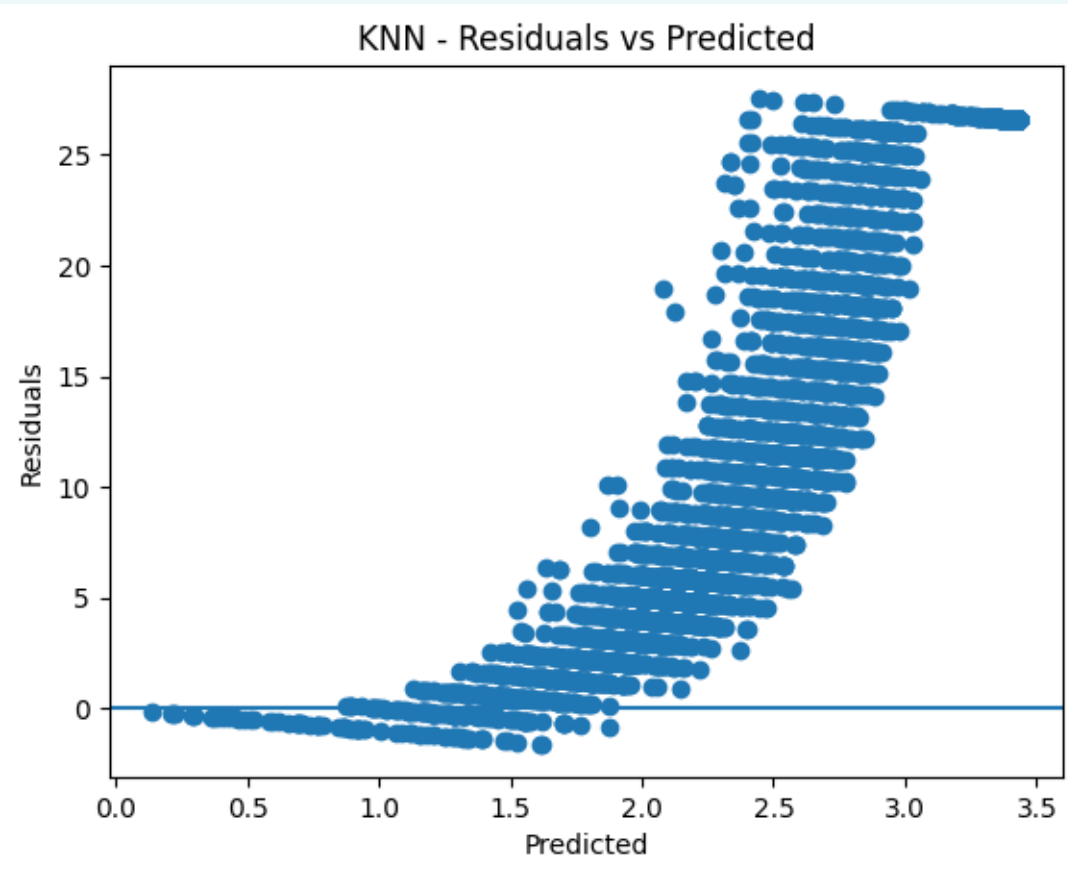


الرسمه بتوضحلنا مقارنة بين القيم الحقيقية والقيم اللي توقعها نموذج KNN كل نقطة بتمثل قيمة حقيقية (Actual) مقابل توقع النموذج إليها (predict)

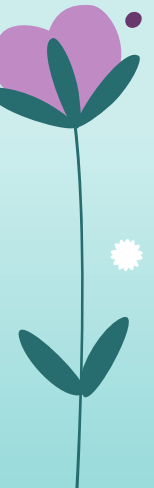
ملاحظ إن التوقعات بشكل عام قريبة من القيم الحقيقية، بس مع زيادة القيم الحقيقية بصير في فرق أكبر شوي، وهذا يعني إن أداء النموذج منيح بالقيم الصغيرة وأضعف شوي بالقيم الكبيرة



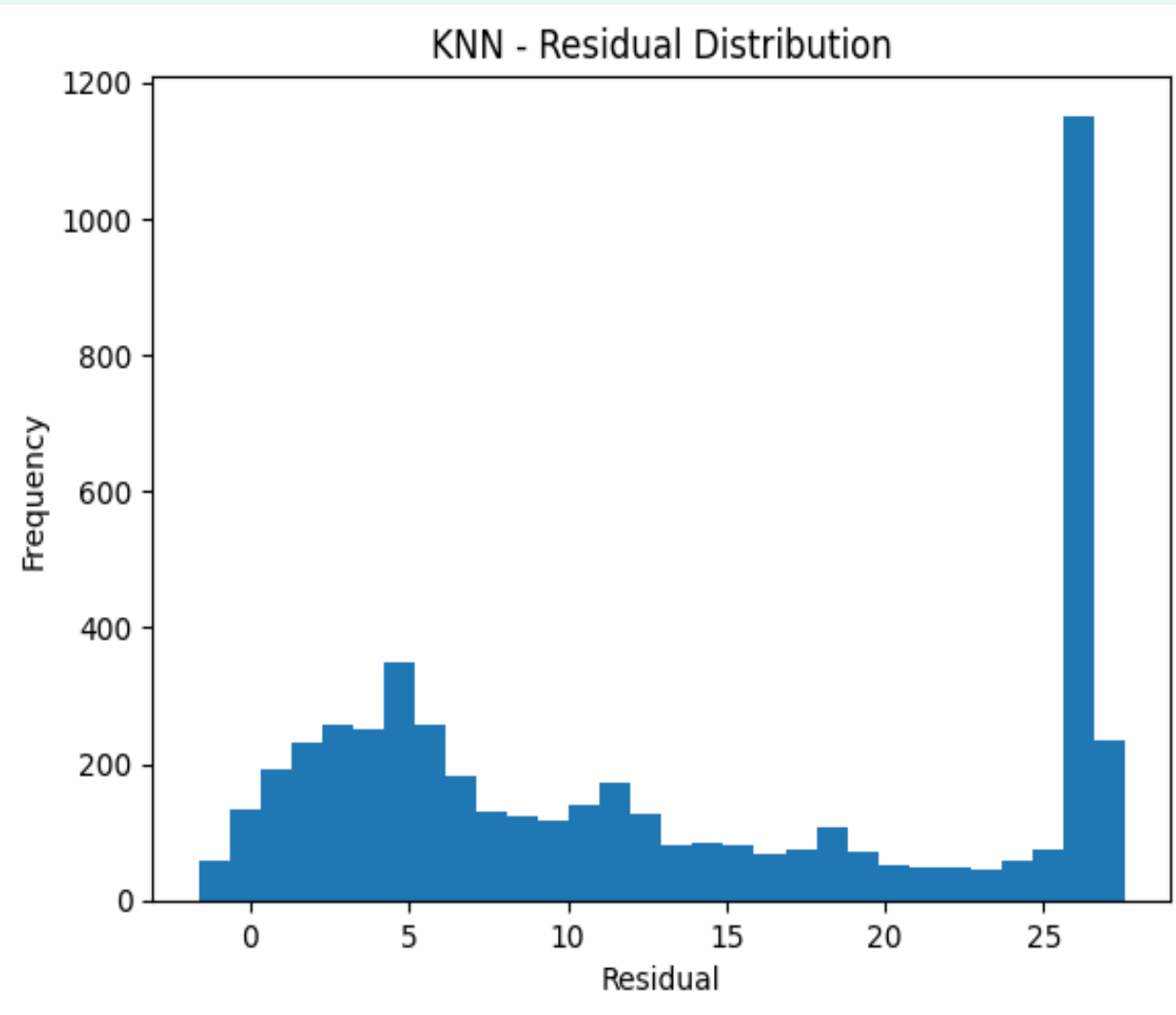
KNN - Residuals vs Predicted



هاي الرسمة بتفرجي الفرق بين القيم المتوقعة والقيم الحقيقية بنموذج KNN لما تكون القيم المتوقعة صغيرة، بنلاحظ إنو الأخطاء قليلة وقريبة من الصفر، يعني النموذج شغله منيح بهالحالات. بس كل ما تكبر القيم، الخطأ بكبر وبصير أوضح، وهاد يعني إنو النموذج ما يكون دقيق كثير مع القيم الكبيرة. بشكل عام، زي ما تعلمنا بنموذج KNN أدائه مقبول مع القيم الصغيرة، بس دقته بتضعف لما القيم تكبر.

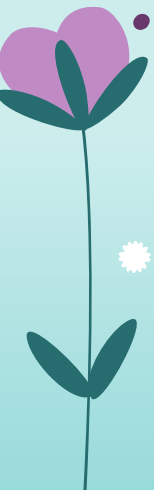


KNN - Residual Distribution

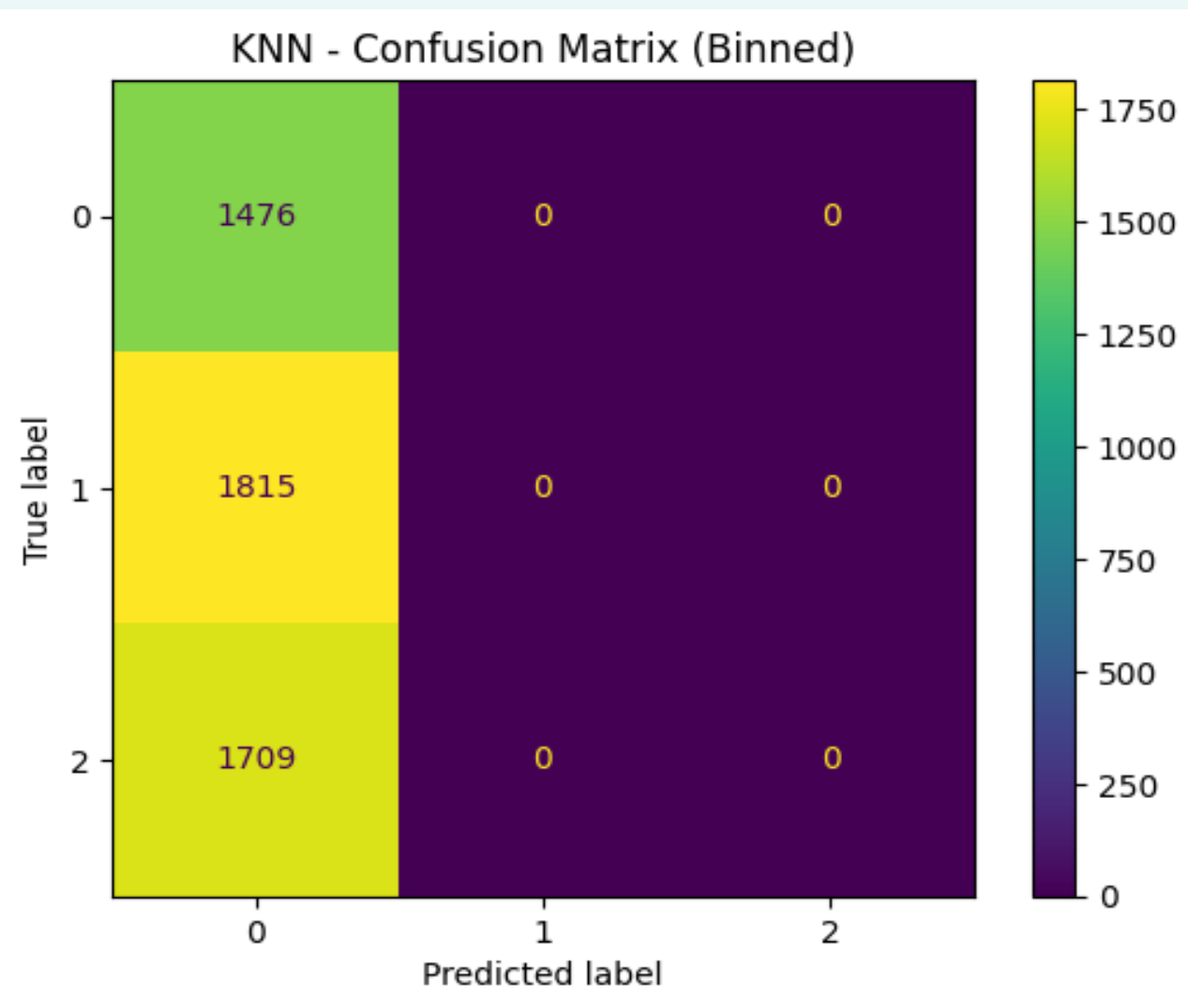


هذه الرسمة بتفرجي توزيع الأخطاء (Residuals) بنموذج KNN

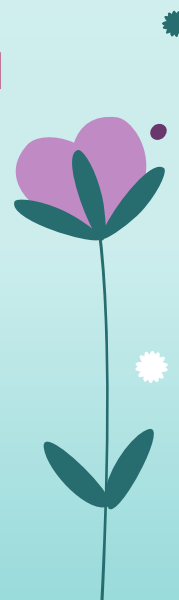
يعني قديش الفرق بين القيمة الحقيقية والقيمة اللي النموذج توقعها. بنلاحظ إنه في جزء كبير من الأخطاء قريب من الصفر، وهاد إشي منيح لأنه يعني إنه النموذج غالبًا توقعاته قريبة من الحقيقة. بس بنفس الوقت في أخطاء كبيرة، خاصة بالقيم العالية، وهاد بوضح إنه النموذج أحيانًا بغلط كثير لما تكون القيم كبيرة. بشكل عام، النموذج أداءه مقبول



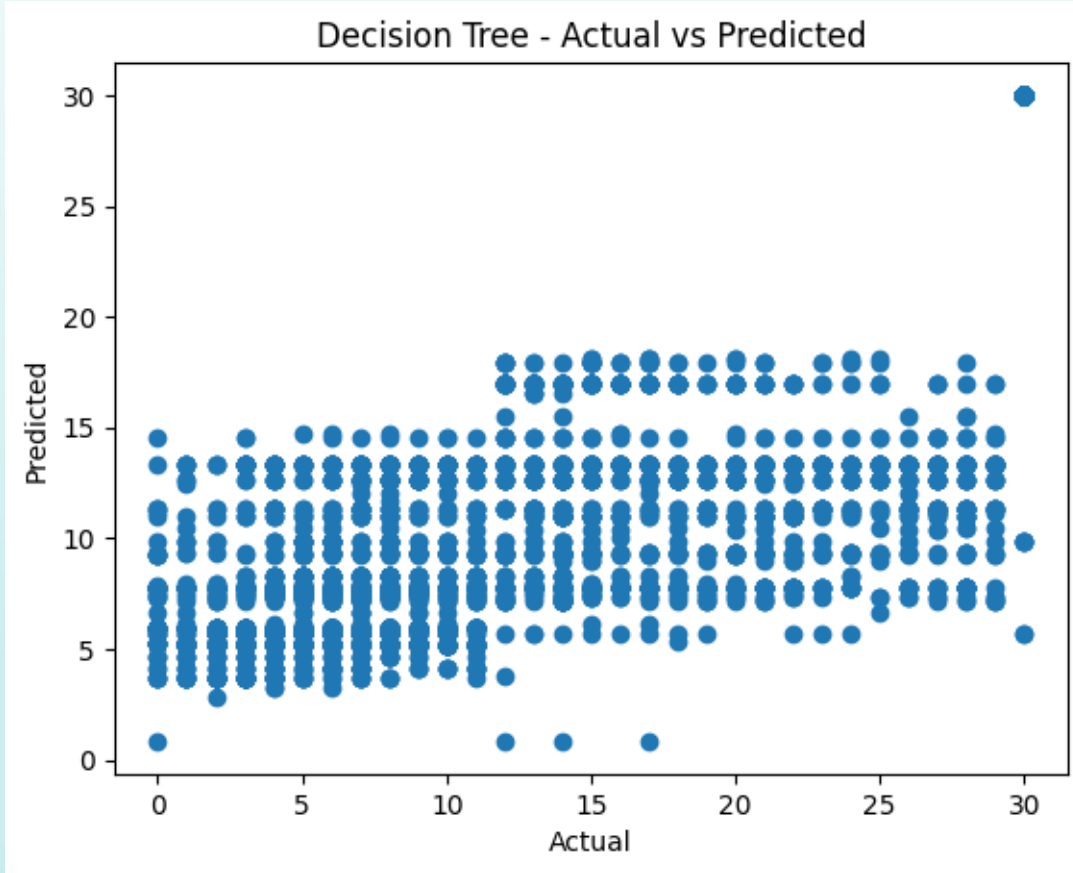
KNN - Confusion Matrix (Binned)



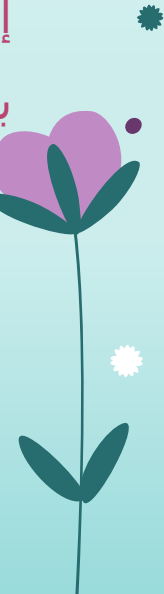
هناك الرسمة بتوضح إنو نموذج KNN
أغلب الوقت بتوقع نفس الفئة، حتى لو
القيم الحقيقية كانت مختلفة. يعني
النموذج ما قدر يفرق منيح بين الفئات،
وخصوصًا بالقيم العالية. هاد بيدل إنو أداء
KNN بهالتصنيف مش قوي، وبغلط لما
البيانات تكون متنوعة



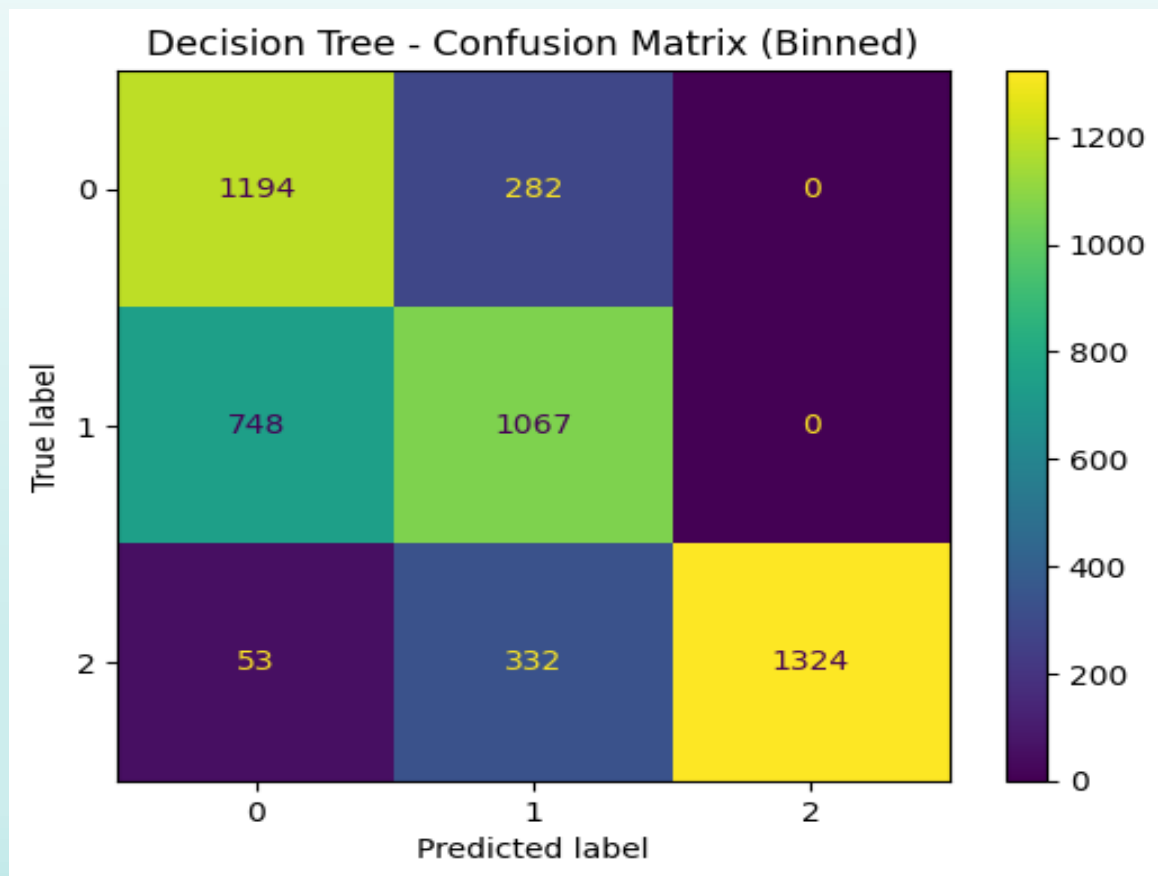
Decision Tree - Actual vs Predicted



هاي الرسمة بتفرجينا مقارنة بين القيم الحقيقية والقيم
اللي نموذج Decision Tree توقعها.
بنلاحظ إنو التوقعات بشكل عام قريبة من القيم الحقيقية،
بس في تشتت واضح خاصة عند القيم الكبيرة. هاد يعني
إنو النموذج أدائه منيح بشكل عام، بس أحيانًا بغلط وما
بكون دقيق 100%



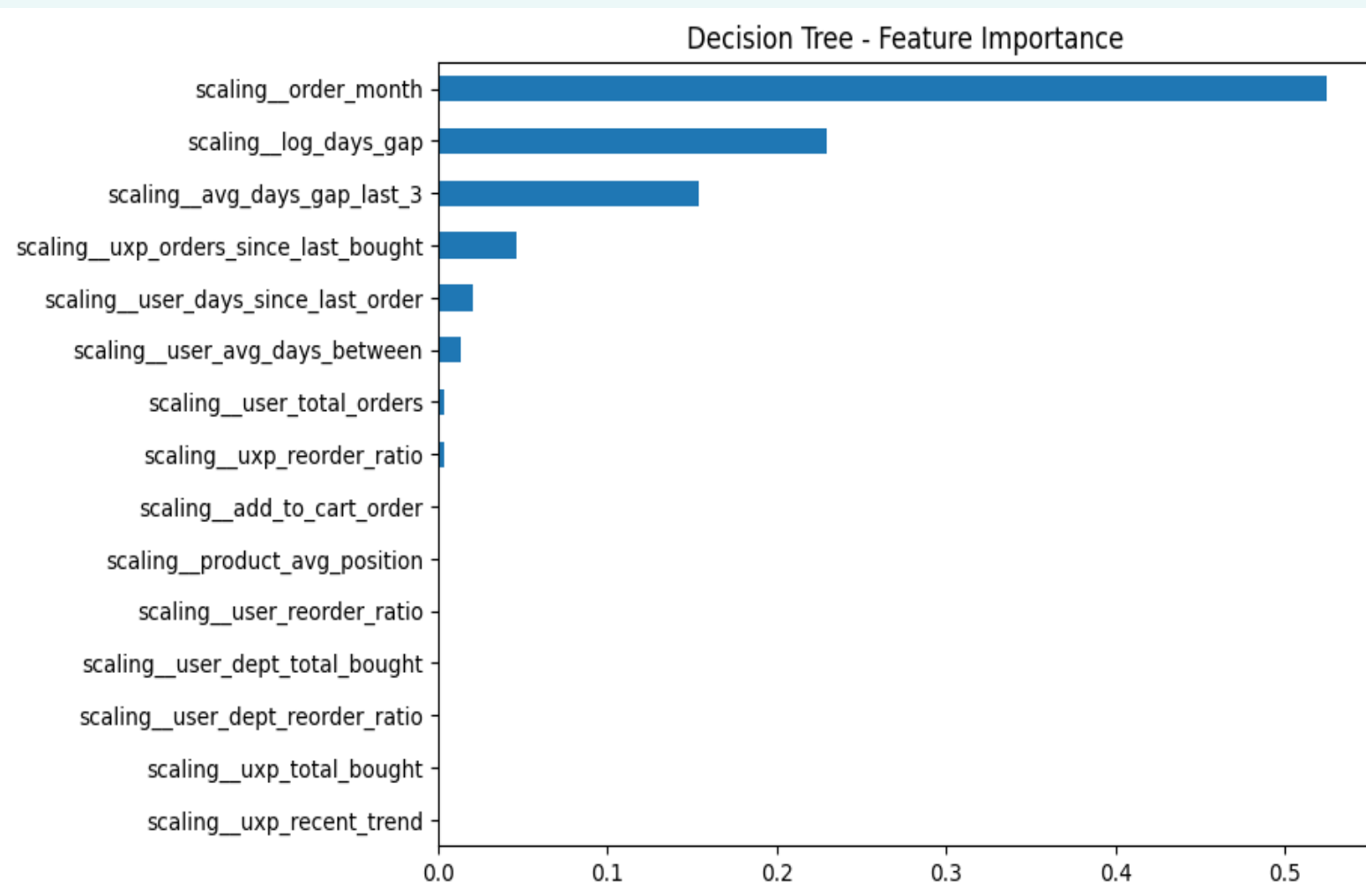
Decision Tree - Confusion Matrix (Binned)



هاي المصفوفة بتوضح أداء نموذج Decision Tree بعد ما قسمنا القيم لفئات. بنلاحظ إنو أغلب القيم موجودة على القطر، يعني النموذج غالبًا بتوقع الفئة الصح. بس في شوية خرابطة بين الفئات، خاصة بين الفئة 0 و1، وهذا يعني إنو أحيانًا النموذج بلخبط بينهم. بشكل عام، أداء النموذج منيح، بس مش مثالي

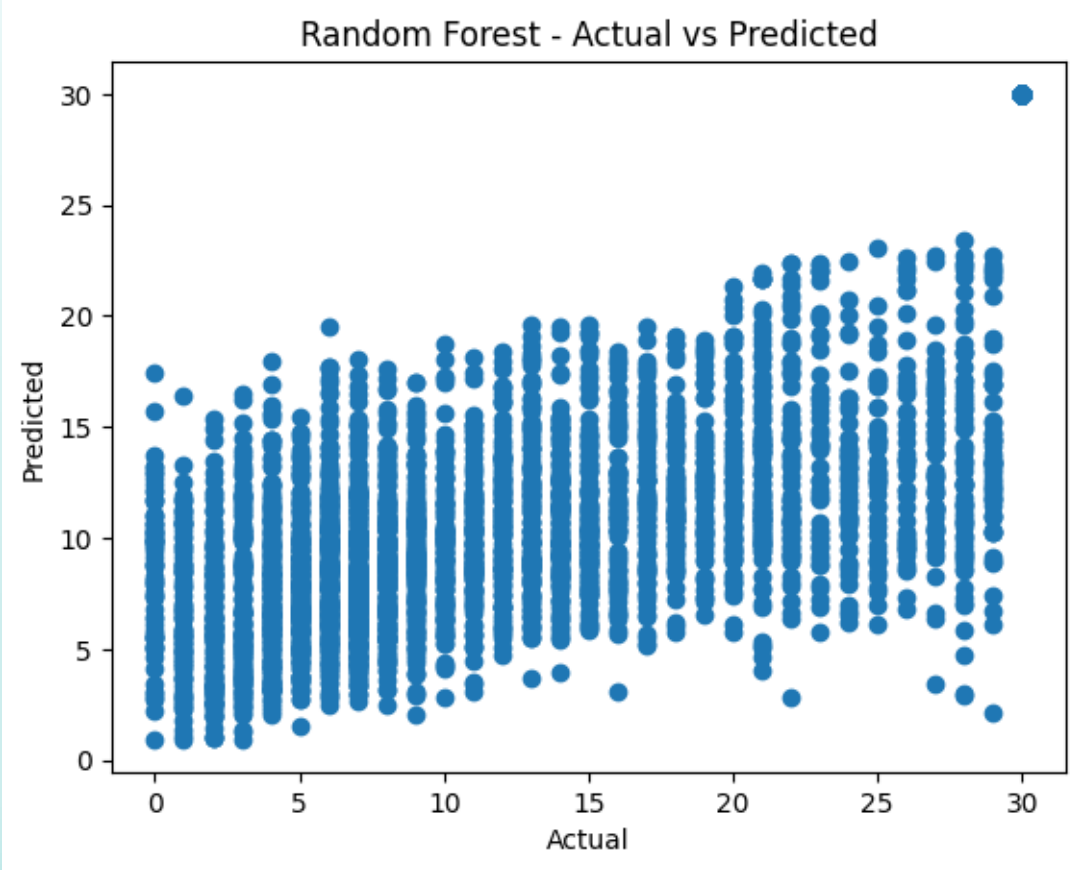


Decision Tree - Feature Importance

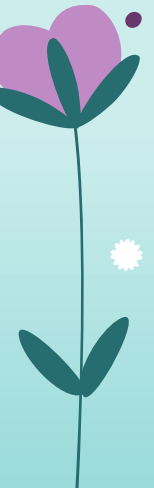


هاي الرسمة بتفرجينا أي features كان
إلها التأثير الأكبر بقرارات نموذج Decision
Tree. بنلاحظ إنو scaling_order_month هو
أكثر إشي أثر بالنموذج، يعني وقت الطلب كان
عامل مهم بالتوقع. بعده إجا عدد الأيام بين
الطلبات، وهذا بوضح إنو سلوك المستخدم من
ناحية التكرار مهم كمان. باقي المتغيرات تأثيرها
أقل، يعني النموذج ما اعتمد عليها كثير. بشكل
عام، النموذج برکز على الوقت وتكرار الشراء أكثر
من أي إشي ثاني

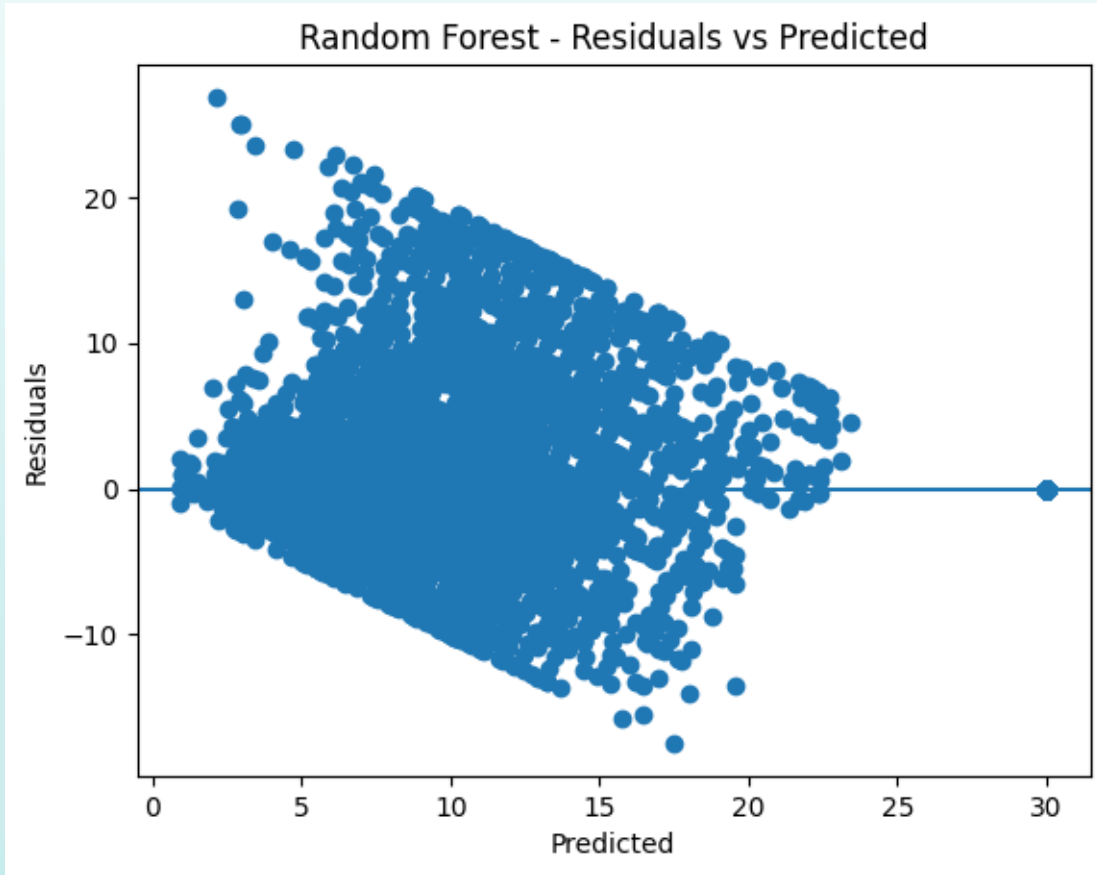
Random Forest - Actual vs Predicted



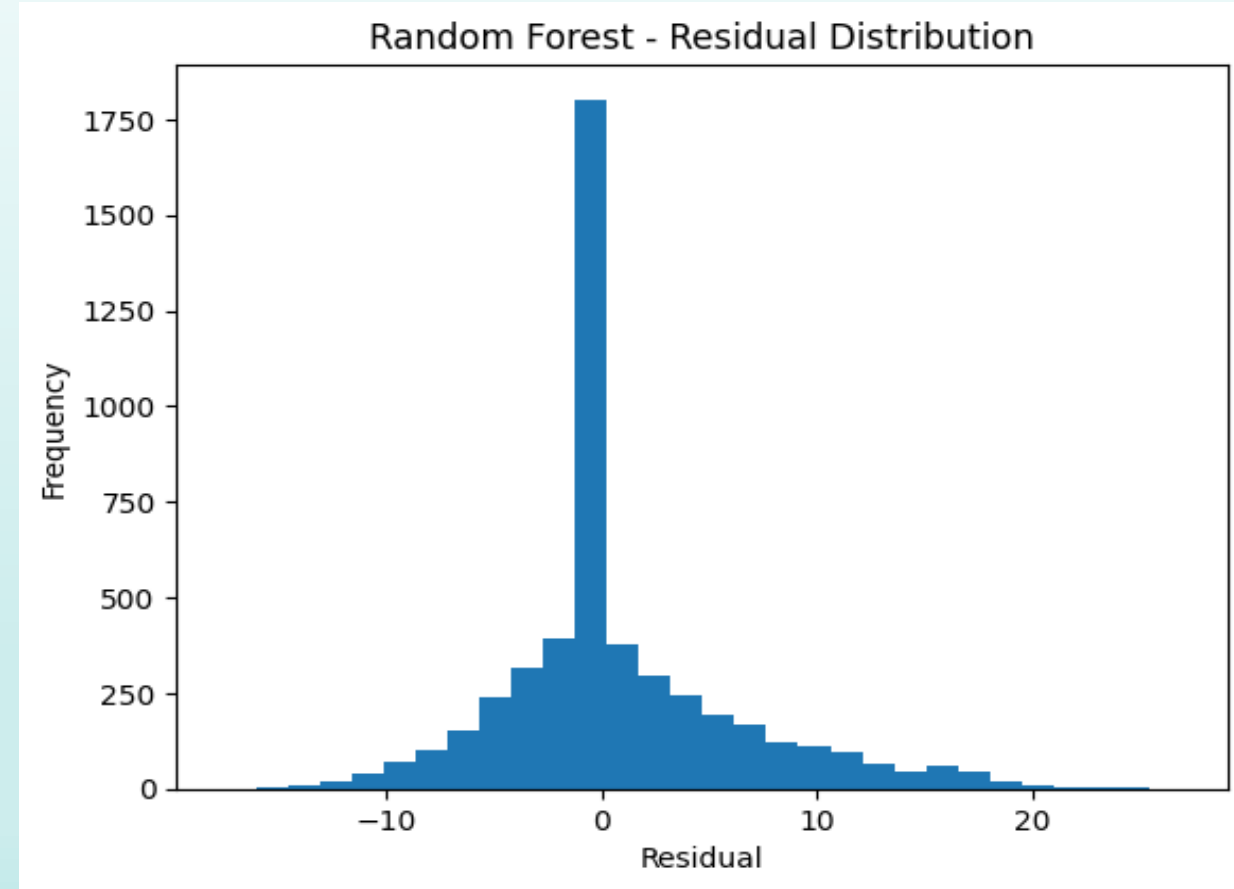
هاي الرسمة بتفرجيننا مقارنة بين القيم الحقيقية والقيم اللي توقعها نموذج Random Forest بنلاحظ إنو أغلب النقاط قريبة من بعض وعلى نفس الاتجاه، وهذا يعني إنو النموذج عم يتوقع بشكل منيح بشكل عام. كل ما كانت القيم الحقيقية أكبر، التوقعات كمان بتكبر، وهذا إشي إيجابي. طبعًا في شوية تشتت وأخطاء، بس بشكل عام أداء النموذج جيد وأحسن من النماذج الأبسط



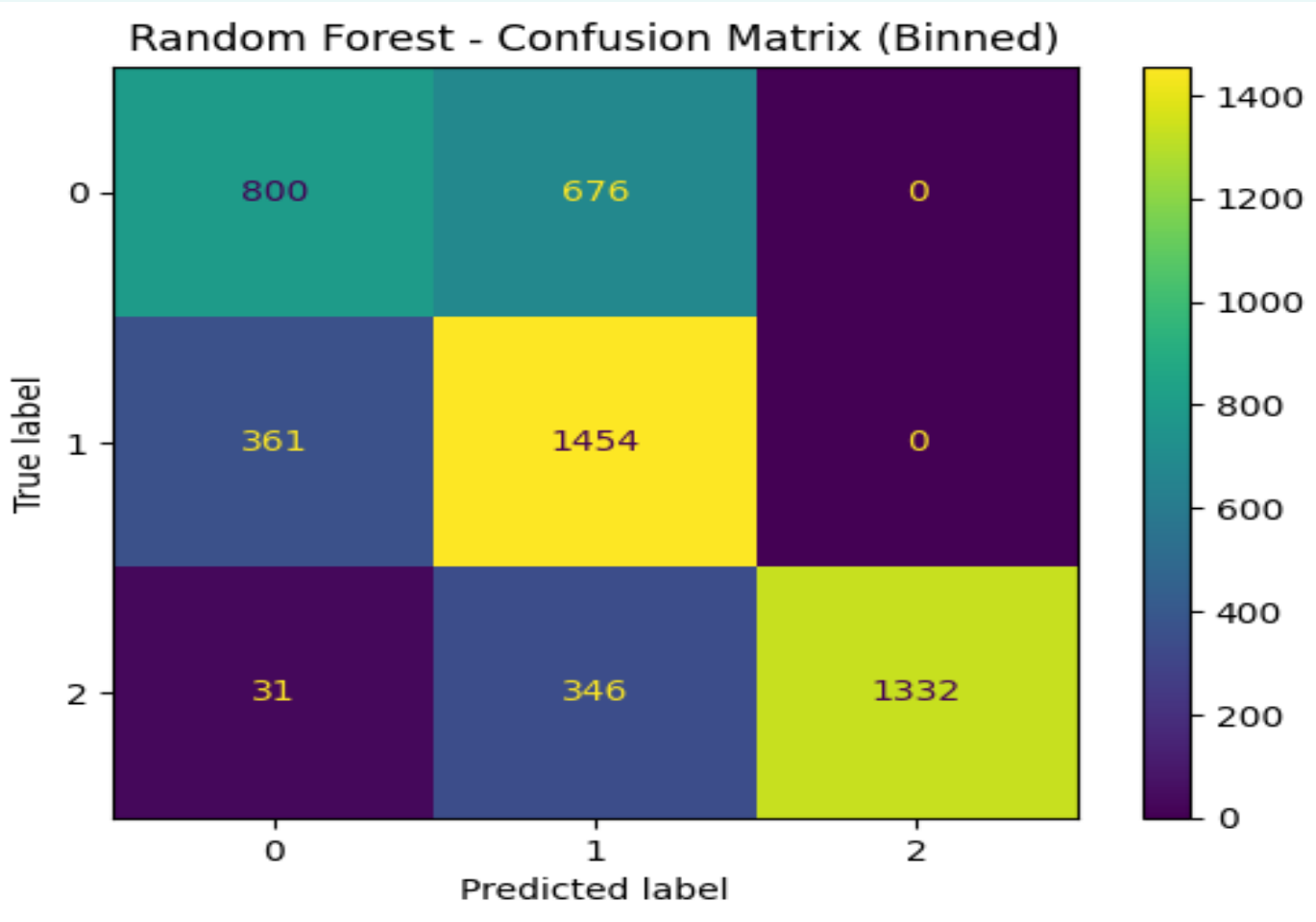
Random Forest - Residuals vs Predicted



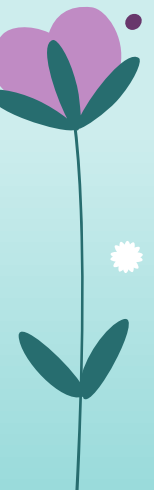
Random Forest - Residual Distribution



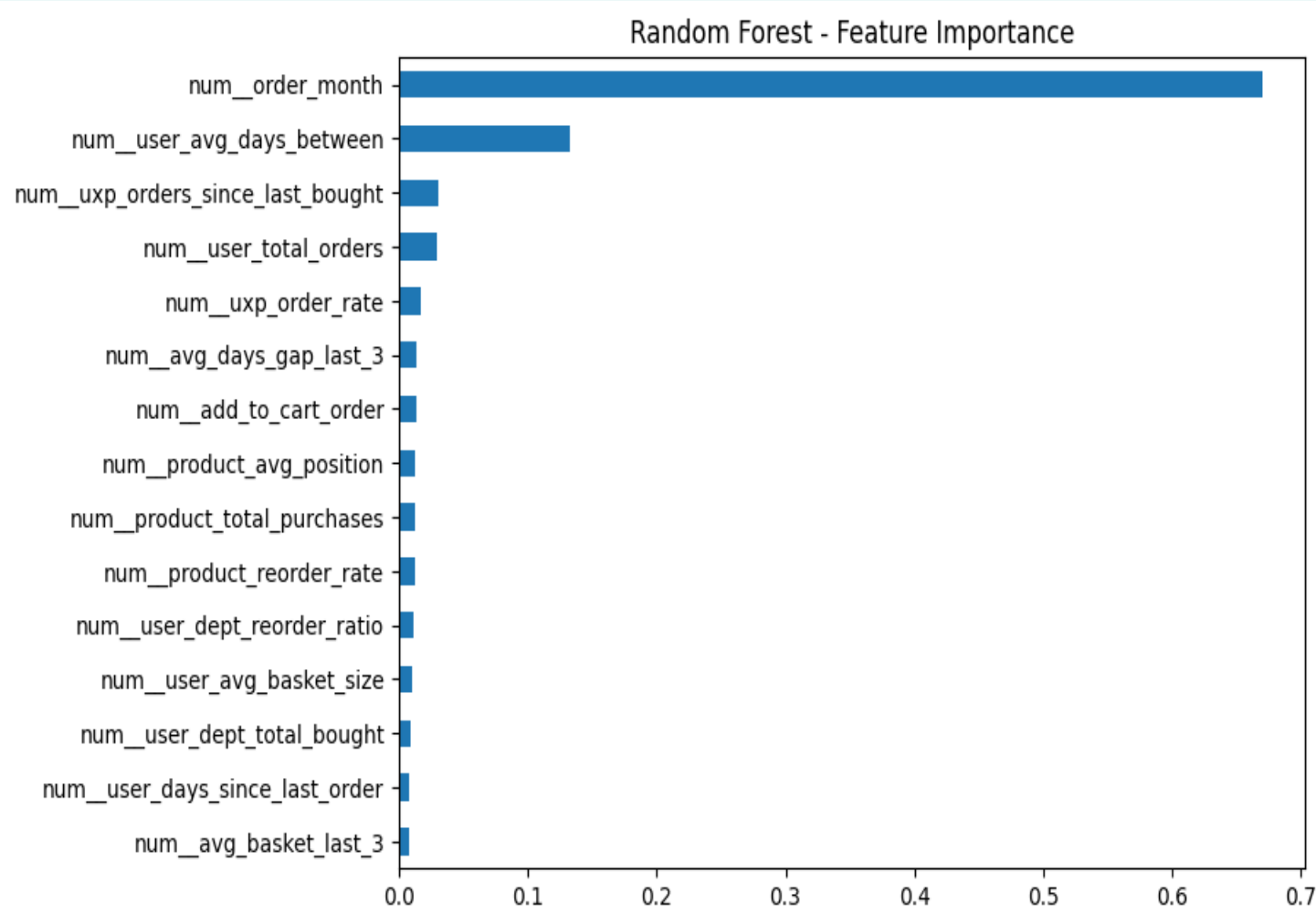
Random Forest - Confusion Matrix (Binned)



هناك الرسمة بتفرجينا ال Confusion Matrix لنموذج Random Forest بعد ما قسمنا القيم لفئات. بنلاحظ إنو أغلب القيم موجودة على القطر، يعني النموذج غالبًا بتوقع الفئة الصح. في شوية لخبطة بين الفئات القريبة من بعض، بس بشكل عام النموذج أداؤه منيح وبفرق بين الفئات بشكل أفضل من النماذج السابقة



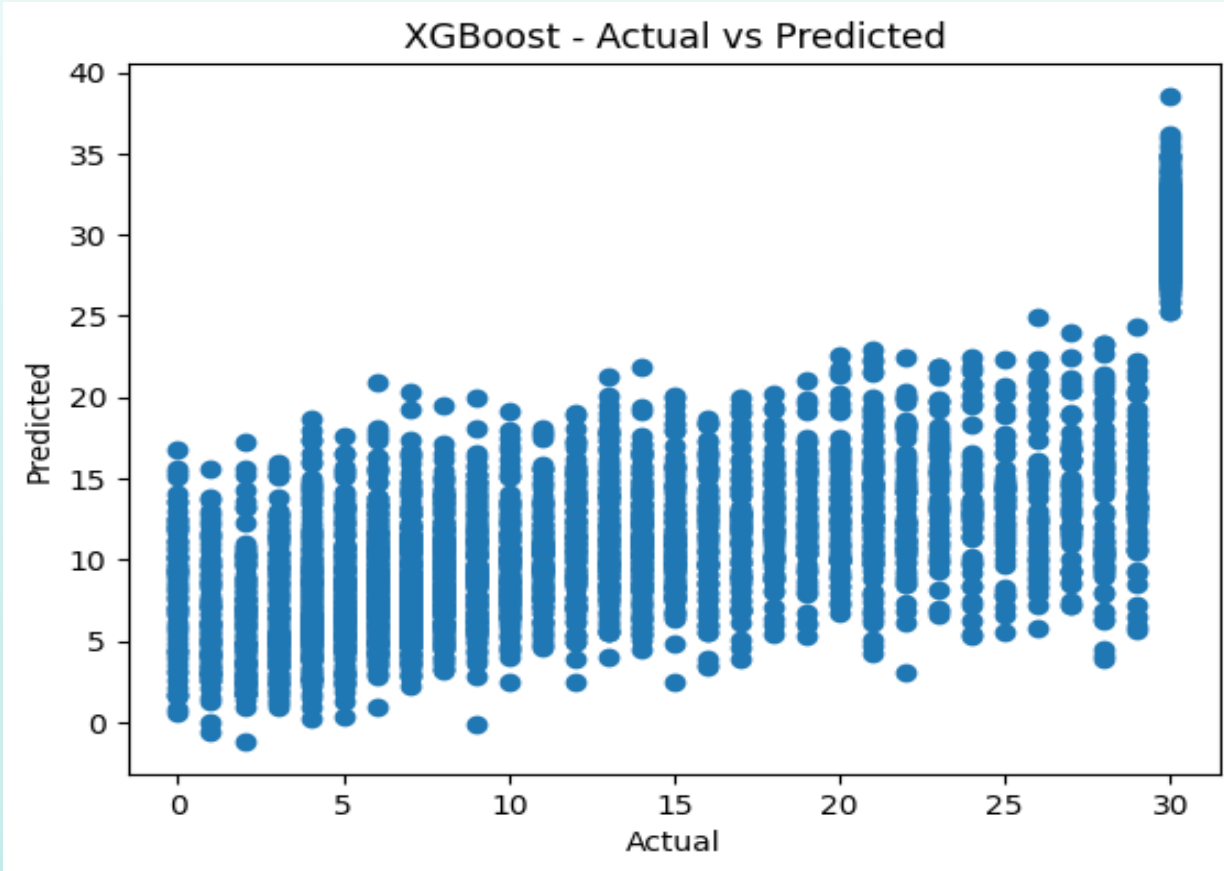
Random Forest - Feature Importance



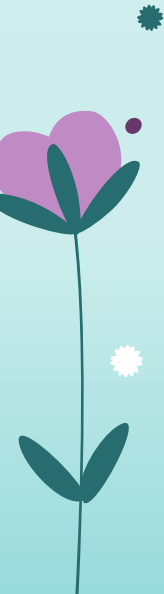
هاي الرسمة بتفرجينا قديش كل متغير إله
تأثير على توقعات نموذج Random Forest
واضح إنو أكثر إشي بأثر هو
num_order_month، يعني وقت الطلب إله
دور كبير بالنتيجة. بعده بييجي average days
between orders، يعني كل ما المستخدم
بطلب بفترات قريبة أو بعيدة يفرق بالتوقع.
باقي المتغيرات تأثيرها أقل، بس لسه إله دور
بسيط. بشكل عام، النموذج بعتمد أكثر على
وقت وسلوك الطلب أكثر من باقي التفاصيل



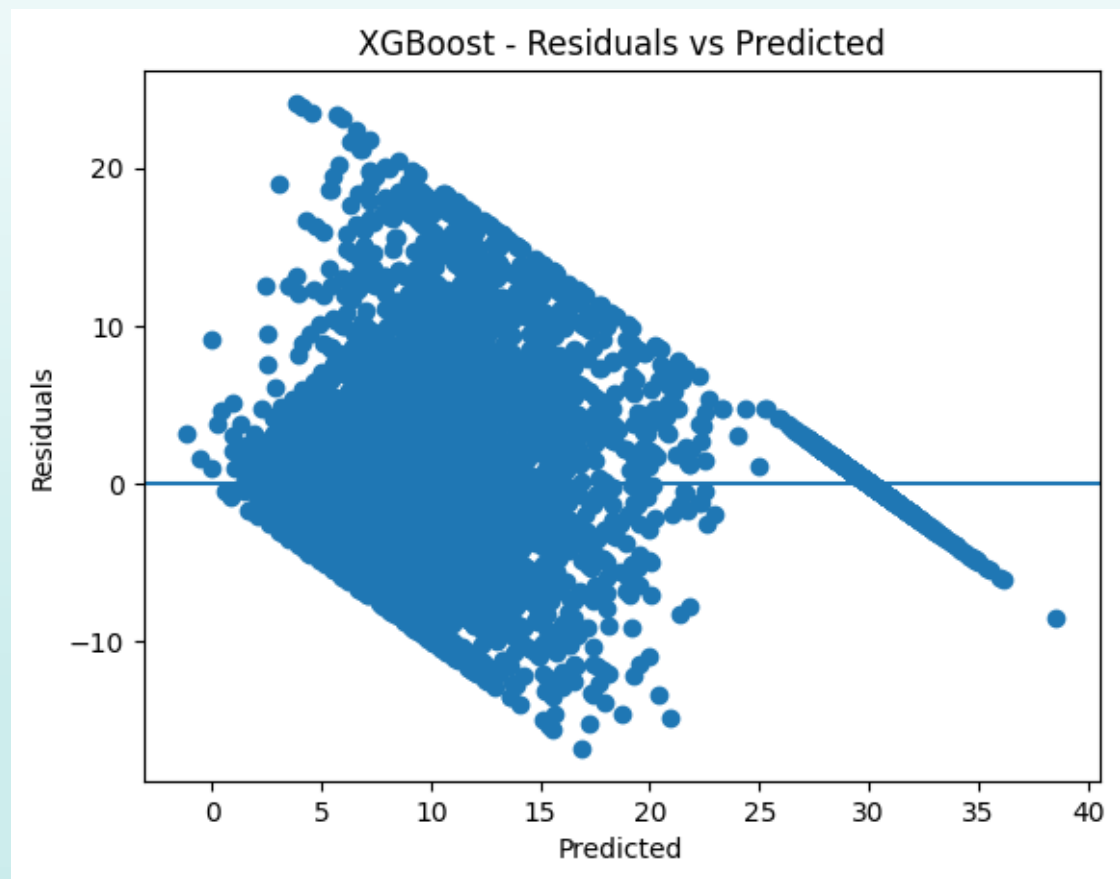
XGBoost - Actual vs Predicted



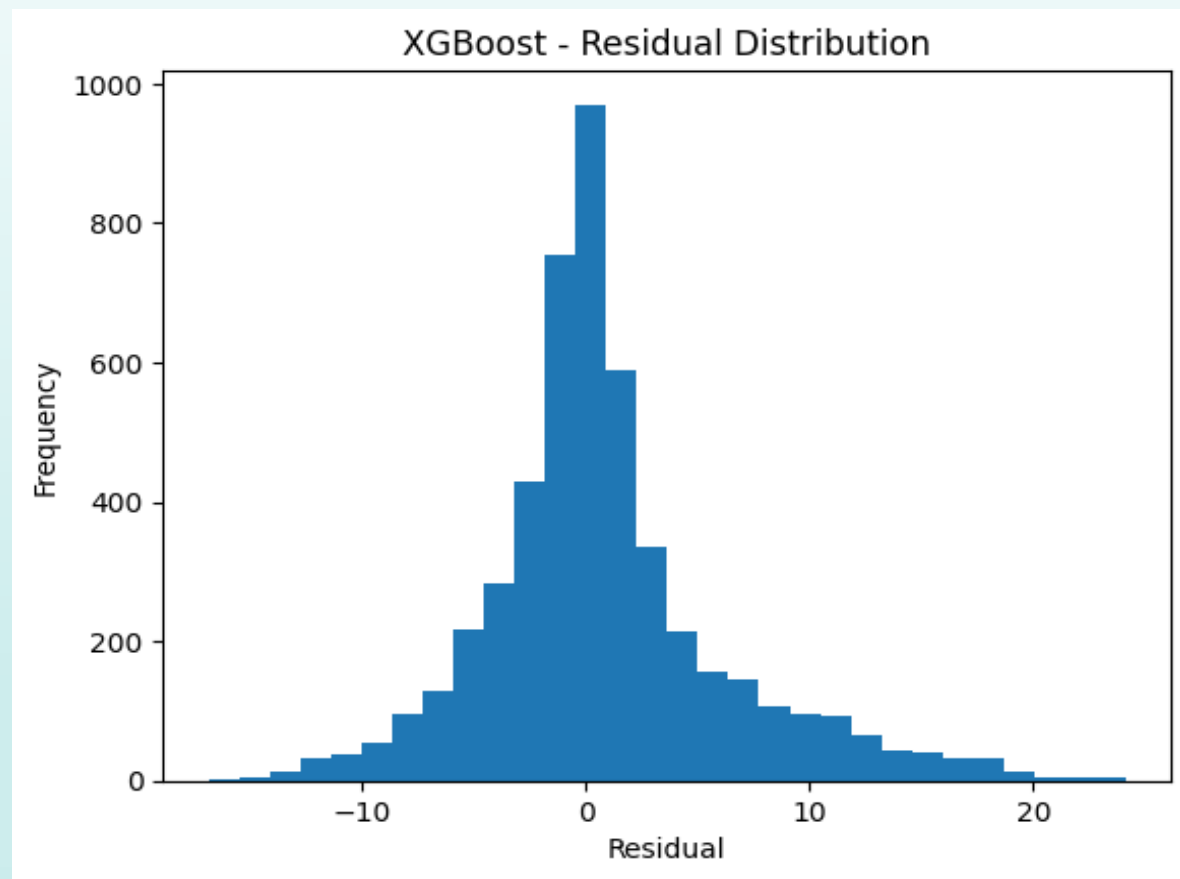
هاي الرسمة بتوضح الفرق بين القيم الحقيقية والقيم اللي توقّعها نموذج XGBoost بشكل عام بنشوف إنو التوقعات قريبة من القيم الحقيقية، خصوصًا بالقيم المتوسطة، وهذا إشي منيح. بس بالقيم العالية، في تشتّت وأحيانًا النموذج ببالغ أو بقصر بالتوقع. يعني النموذج أدائه منيح بشكل عام، بس لسه عنده صعوبة شوي مع القيم الكبيرة



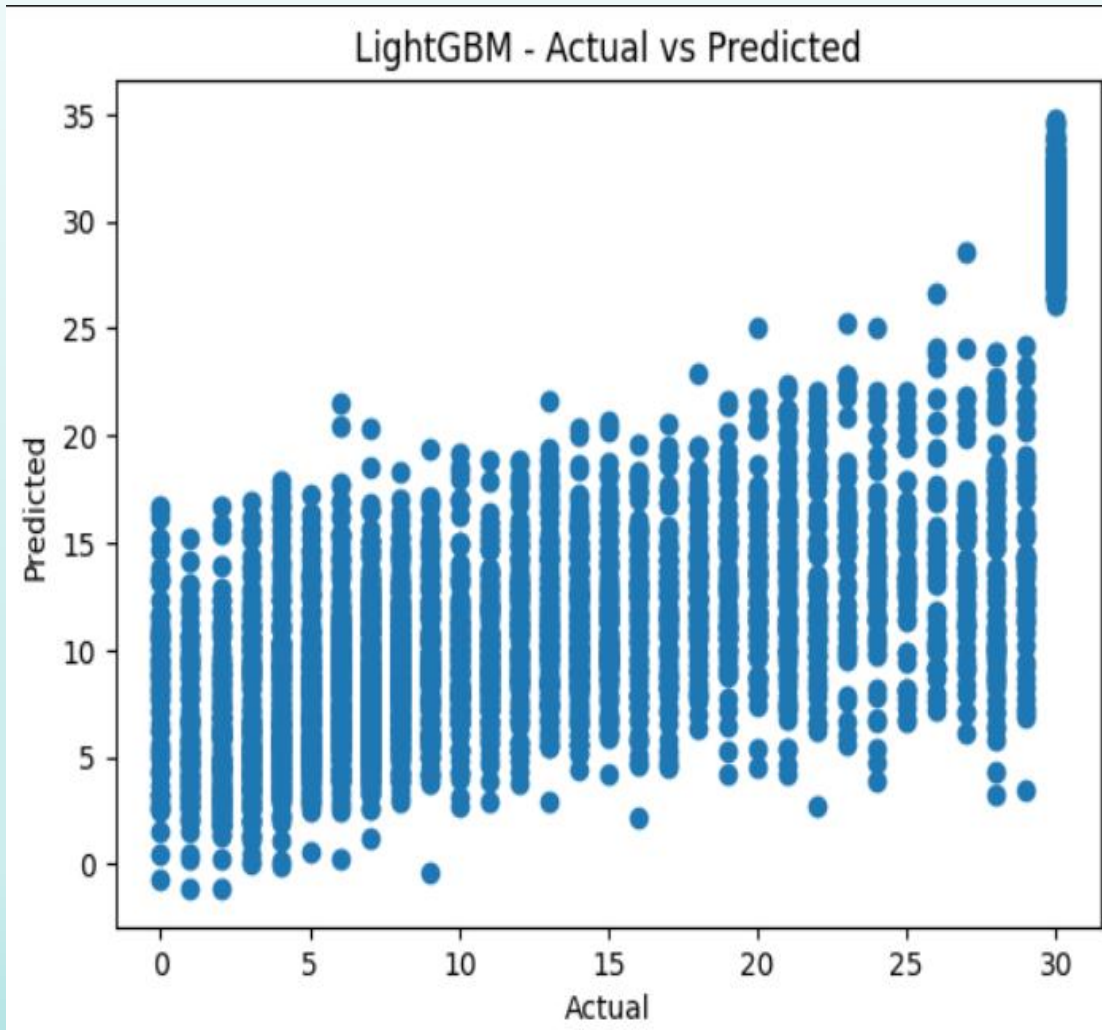
XGBoost - Residuals vs Predicted



XGBoost - Residual Distribution

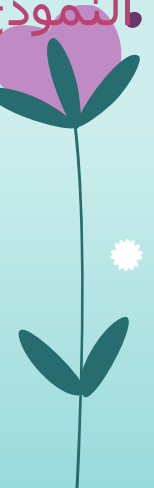


LightGBM - Actual vs Predicted

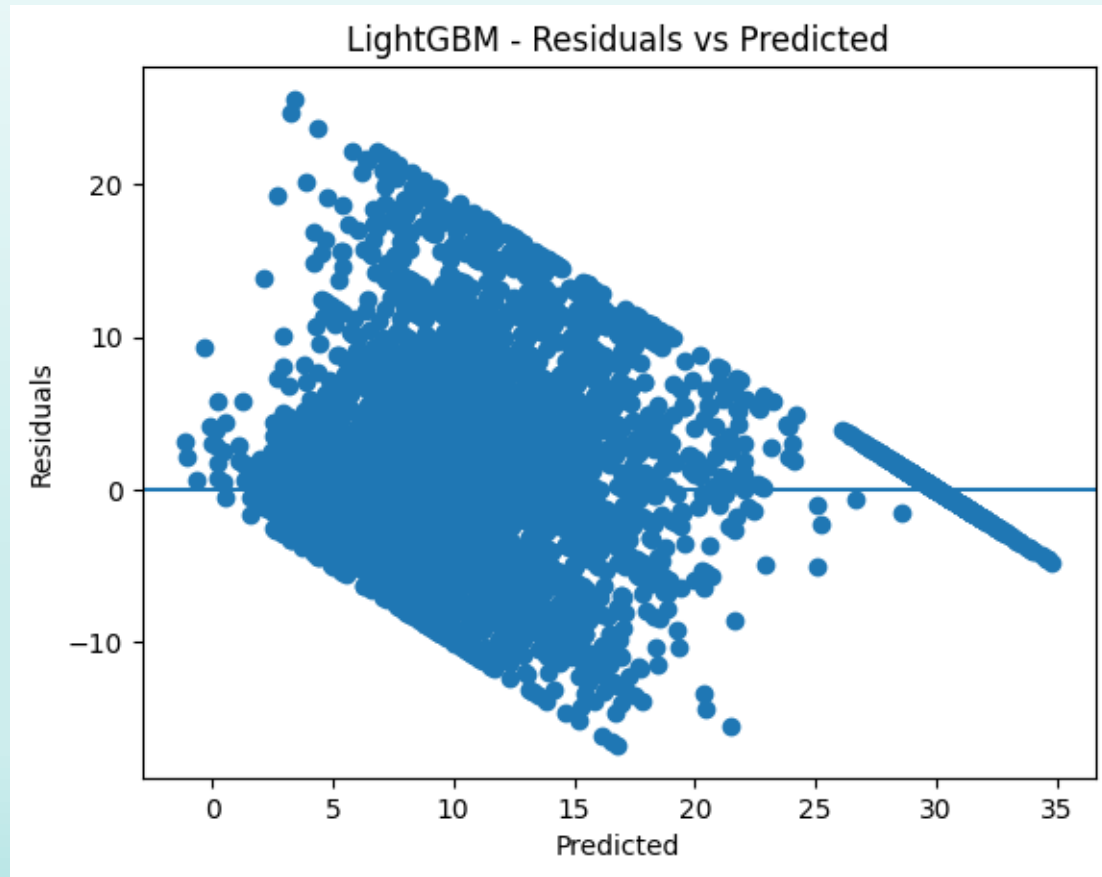


هاي الرسمة بتفرجي مقارنة بين القيم الحقيقية والقيم
اللي توقّعها نموذج LightGBM

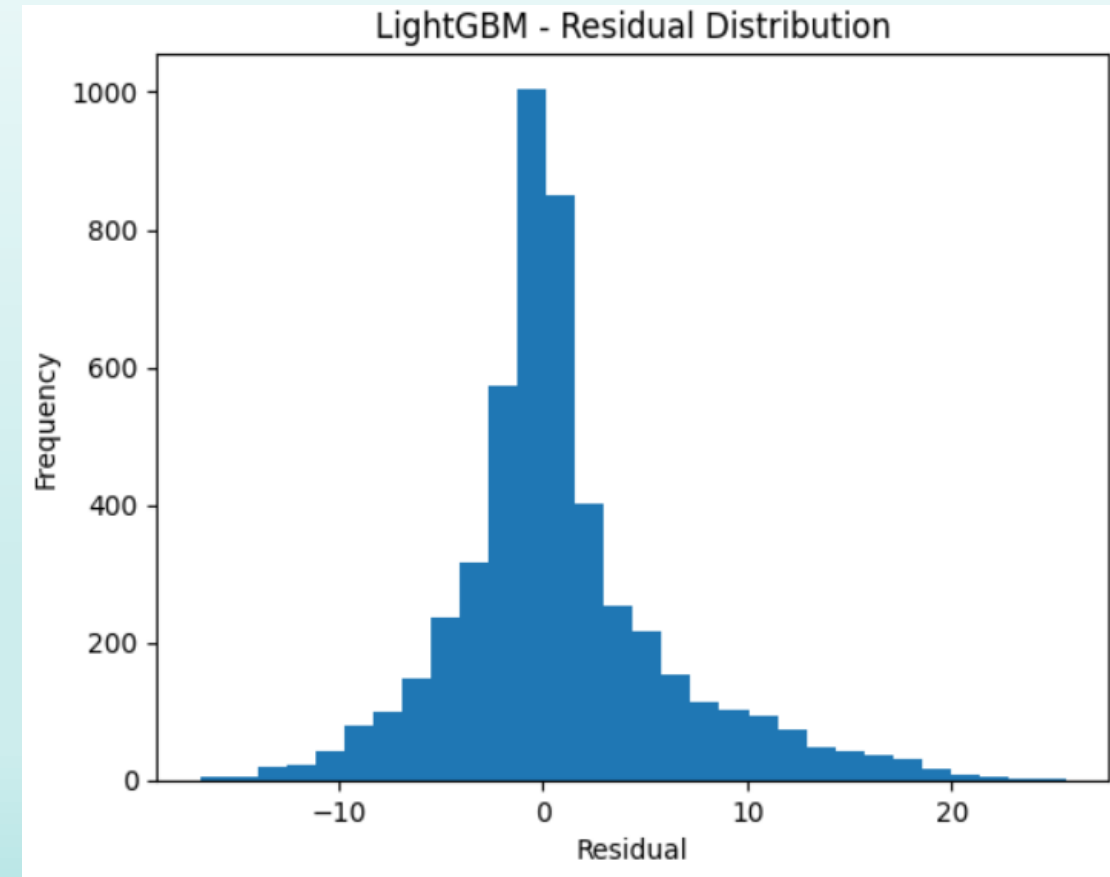
بشكل عام في علاقة واضحة بينهم، يعني كل ما القيمة
الحقيقية تكبر، التوقع كمان بكبر، وهاد إشي منيح. التوقعات
بالقيم المتوسطة قريبة من الحقيقة، بس بالقيم العالية في
تشّتت وأحيانًا النموذج ما بيمسكها بدقة. بشكل عام أداء
النموذج منيح، بس لسه بده تحسين بالقيم الكبيرة



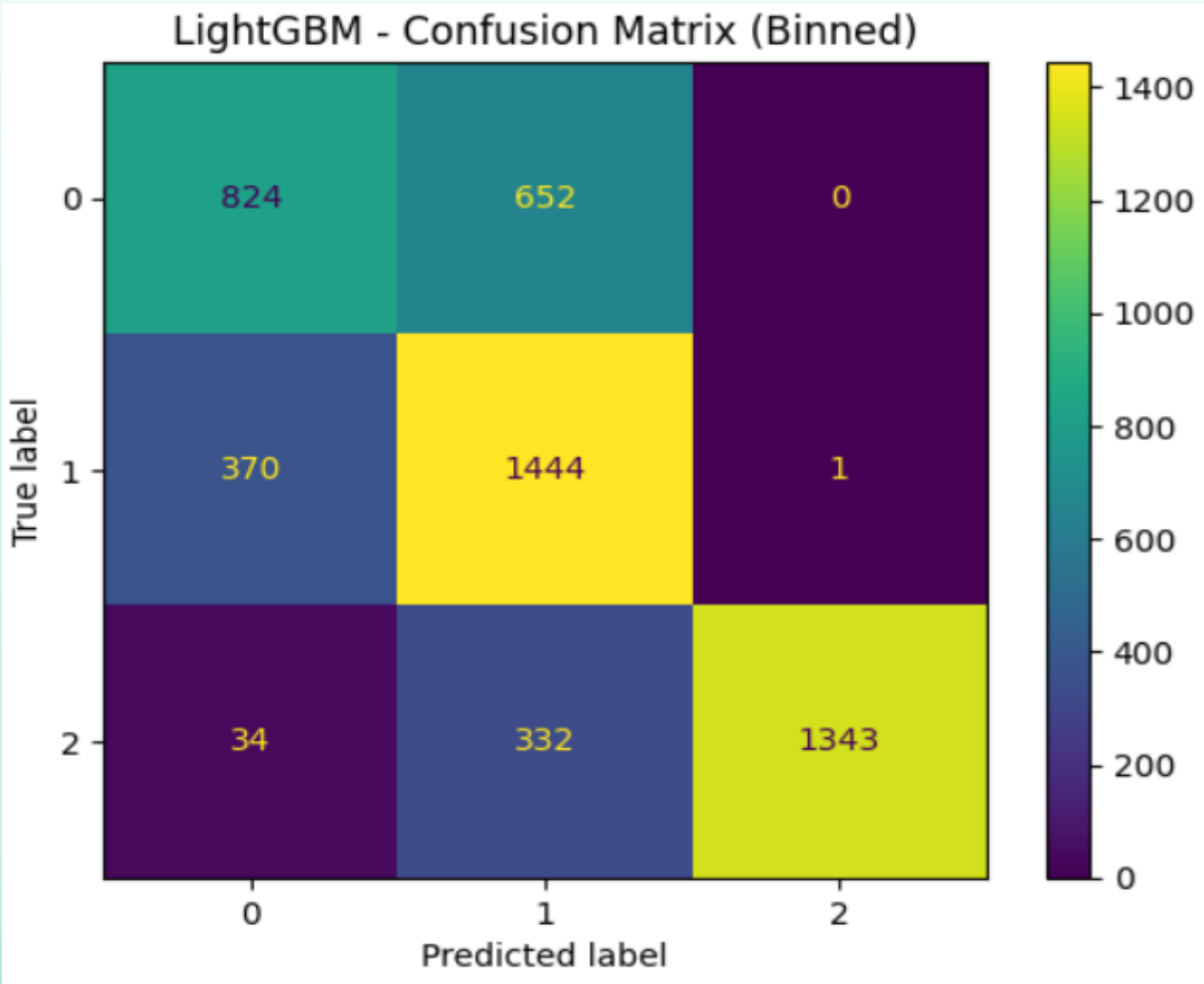
LightGBM - Residuals vs Predicted



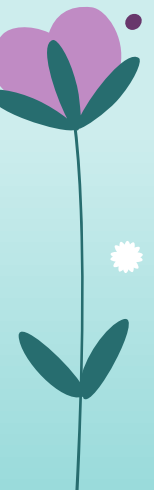
LightGBM - Residual Distribution



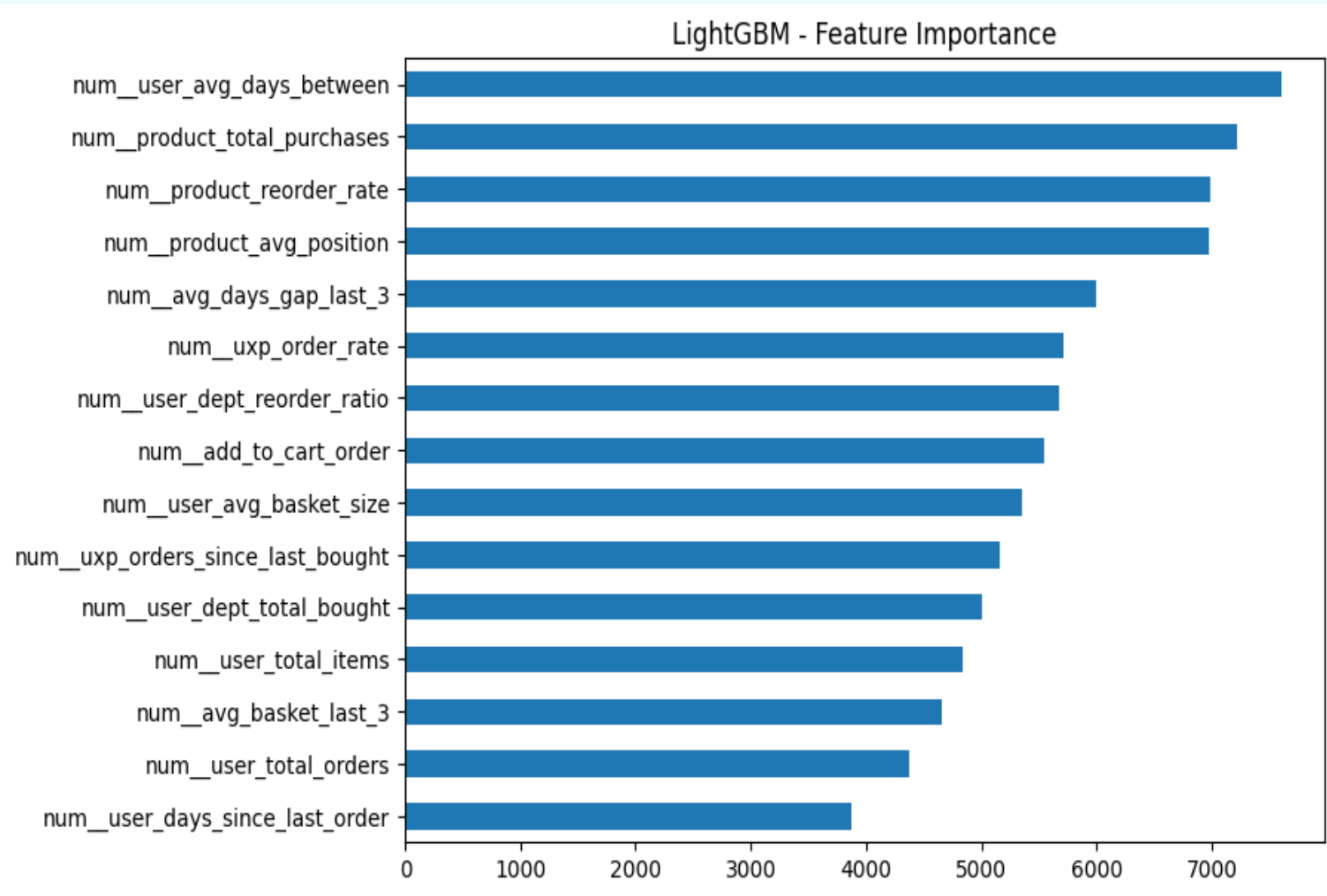
LightGBM - Confusion Matrix (Binned)



هاي المصفوفة بتفرجي كيف نموذج LightGBM صنف القيم بعد ما قسمناهم لفئات. أغلب القيم الصح موجودة على القطر، يعني النموذج غالبًا يعرف يحط القيمة بالفئة الصح. في شوية لخبطة بين الفئة 0 و1، وكمات بين 1 و2، بس بشكل عام النتائج منيحة. يعني النموذج أدائه كويس، بس أحيانًا بخلط بين الفئات القريبة من بعض



LightGBM - Feature Importance



هاي الرسمة بتفرجي أكثر المتغيرات اللي نموذج LightGBM اعتمد عليها بالتوقع. بنلاحظ إنو عدد الأيام بين طلبات المستخدم وعدد مرات شراء المنتج كانوا الأكثر تأثير. باقي المتغيرات إلها تأثير أقل بس لسه بتساعد النموذج. يعني النموذج ركّز أكثر على سلوك المستخدم وتكرار الشراء ليطلع

النتيجة

