

The Final Presentation

Final Listing Price Prediction for Private Used Car Sellers



Group 2

Ji-Soo Kim, Tae-Yoon Kim, Jun-Beom Lee, Jin-Joo Yang, Sung-Hyun Kim

Table of Contents

| | |
|------------------------------------|-------|
| 1. Introduction | _____ |
| 2. Target Audience Analysis | _____ |
| 3. Our Dataset | _____ |
| 4. Data Preprocessing | _____ |
| 5. Model Works | _____ |
| 6. Model Results | _____ |
| 7. Conclusion | _____ |

1. Introduction

News

How much profit do car dealers make on new and used cars? Here's the perception versus reality

- Huge disparity between how much buyers think dealers make and the reality
- Some consumers think dealers are making more than 75 per cent profit per car
- What Car? survey shows average consumer thinks dealers make 10-20 per cent per unit
- Dealers reveal that they really make about seven per cent on new cars

The used car market is a lot stronger with profit margins for dealers around 12 to 15 per cent.

David Kendrick, partner and accountancy UHY Hacker Young, said the figures dealers quoted to Car Dealer are accurate.

He said: 'There is a huge misconception as to how much retailers make on vehicle sales.

'We take on a large number of graduates every year and one of our questions is how much the average dealer makes on a £15k vehicle.

'Answers range from £10k down to those who understand it may be minimal to a loss.

'The issue is that there is such a huge range and misunderstanding. The substantial multi-million pound dealerships certainly don't help that perception. I am not sure how this will ever change.'

Kendrick said the average most dealers make on new and used cars is around £1,000 a car.

Dealers benefiting from the difference between the selling price and the buying price of a used car

➡ **Sellers** have difficulty measuring the price they can actually obtain..😭

1. Introduction

BAIN & COMPANY

Industries ▾ Consulting Services ▾ Digital Insights ▾ About ▾ Careers

Explore 🔍 | 📌

Brief

The Outlook for the European Used Car Market

Growth opportunities may be closer than they appear.

By Roch Baranowski, Eric Zayer, Klaus Stricker, and Ingo Stein


10 min read

←

🖨

📄

🔖

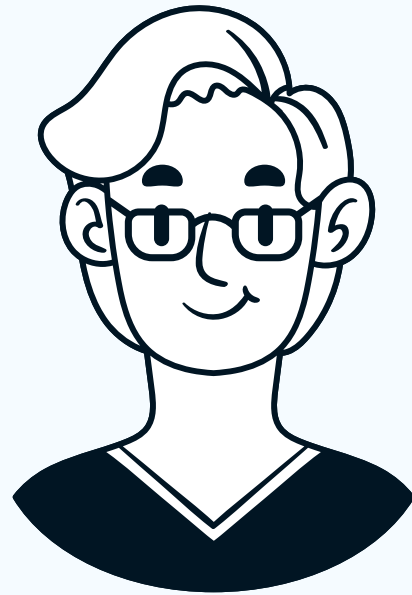


With a total volume of 429 billion euros in 2021, the market for used cars in Europe is similar to the market for new cars. Of the 32 million used cars sold in Europe in 2021, some 44% were sold privately in consumer-to-consumer (C2C) transactions, while 56% were sold by professional retailers, which tend to focus on higher-class, younger cars. Furthermore, used car margins typically beat new car margins for dealers. Overall, the used car market is growing healthily, with a CAGR of 7% between 2015 and 2021.

Private Transactions are not easy to investigate with official statistics

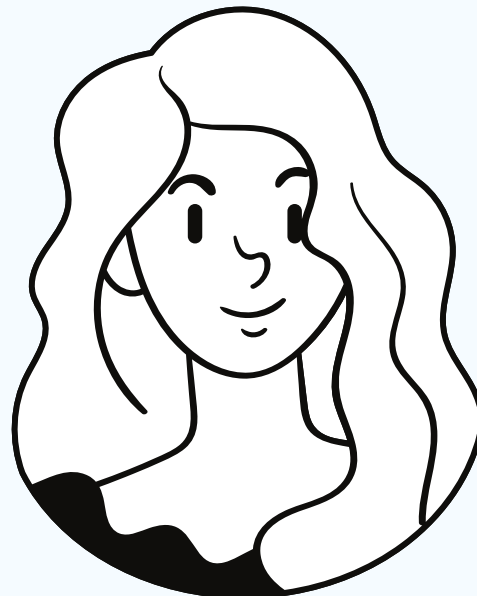
➡ **Buyers** also have difficulty determining whether the sale posted on the used car site is a false sale or not..😭

2. Target Audience Analysis - **Seller** & **Buyer**



Daniel Brühl - **Seller**

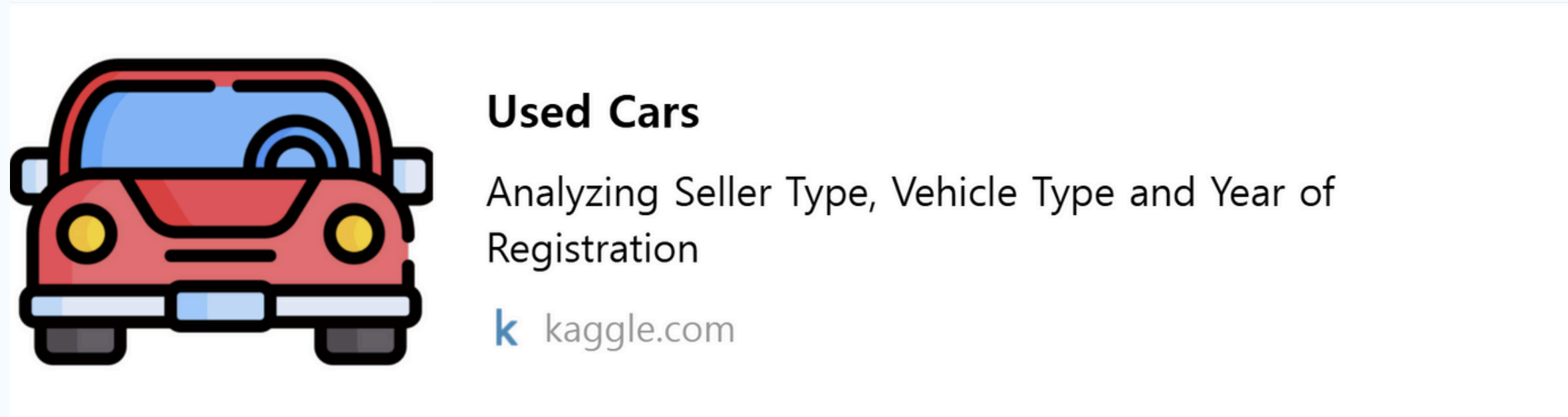
- Trying to sell my original car in order to buy a new car!
- But I'm afraid of the dealer fraud....
- Can't I just **sell this privately** with “Appropriate Price”?



Maria Stein - **Buyer**

- Want to buy used car cheaper because I'm out of budget.
- Then I have to find a used car in private sale, without a dealer...
- How much will be “the common selling price” of my dream car?

3. Our Dataset



21 features

X

233531 data samples



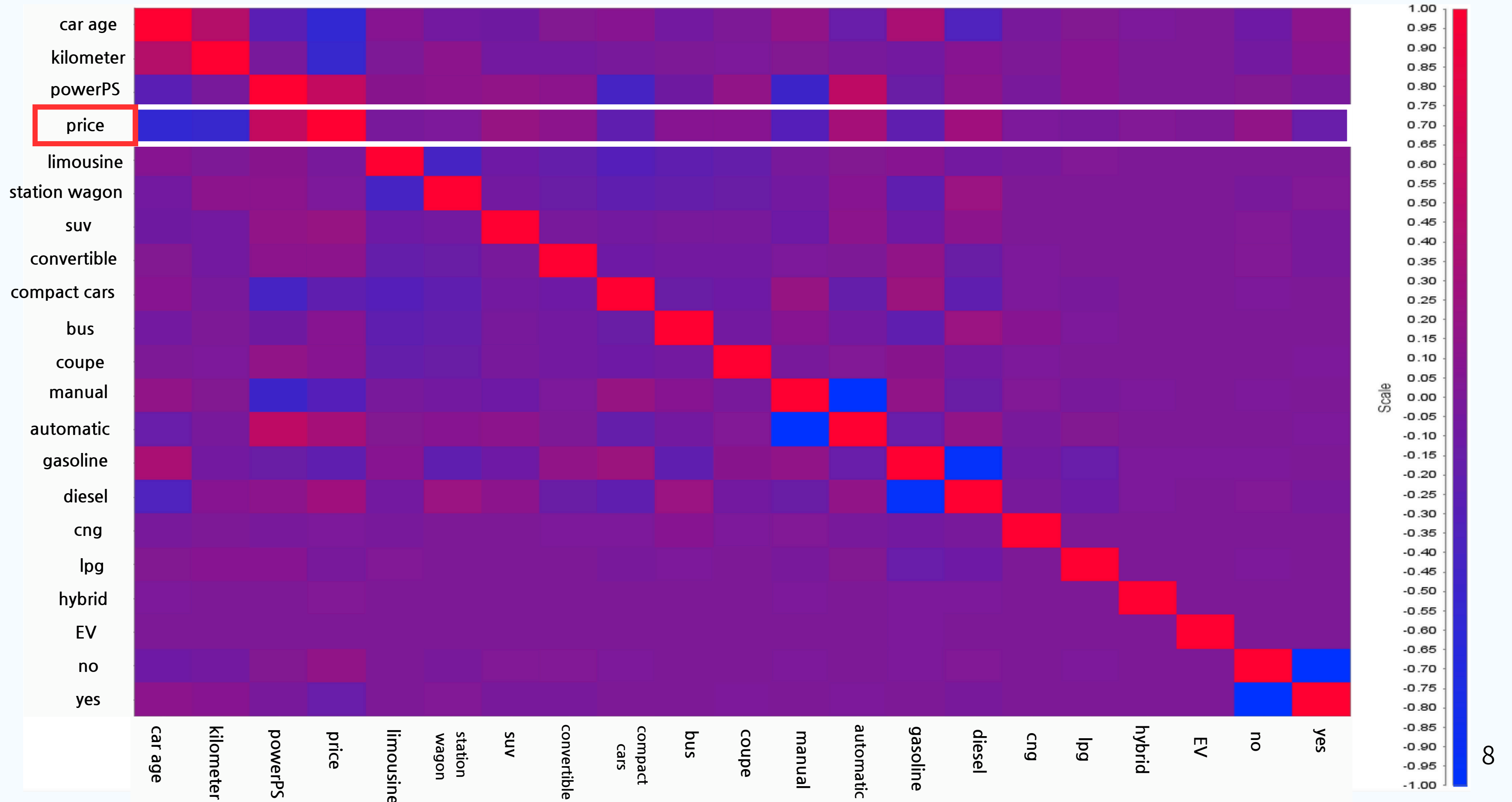
dateCrawled, name, seller,
offerType, price, abtest, vehicleType,
yearOfRegistration, gearbox etc.

Source of data : ebay or other sites where cars can be posted.

3. Our Dataset

| car_brand | car_model | car_age | kilometer | vehicleType | gearbox | powerPS | fuelType | notRepairedDamage | price |
|------------|-----------|----------|-----------|---------------|-----------|---------|----------|-------------------|-------|
| volkswagen | golf | 19 | 150000 | limousine | manual | 75 | gasoline | no | 600 |
| volkswagen | golf | 17.58333 | 150000 | station wagon | manual | 90 | diesel | no | 1459 |
| volkswagen | passat | 7.666667 | 150000 | station wagon | manual | 170 | diesel | no | 5890 |
| volkswagen | golf | 9.75 | 150000 | limousine | manual | 80 | gasoline | no | 4000 |
| volkswagen | touareg | 10.91667 | 150000 | suv | automatic | 224 | diesel | no | 11900 |
| volkswagen | beetle | 12.66667 | 150000 | convertible | manual | 75 | gasoline | no | 5800 |
| volkswagen | fox | 5 | 80000 | compact cars | manual | 60 | gasoline | no | 7800 |
| volkswagen | golf | 4 | 60000 | limousine | manual | 211 | gasoline | no | 19200 |
| volkswagen | golf | 15.83333 | 150000 | limousine | manual | 75 | gasoline | no | 1200 |
| volkswagen | golf | 32.16667 | 150000 | limousine | manual | 75 | gasoline | no | 3799 |
| volkswagen | golf | 6.166667 | 90000 | limousine | manual | 105 | diesel | no | 8990 |
| volkswagen | golf | 21.75 | 150000 | convertible | manual | 90 | gasoline | no | 1750 |
| volkswagen | passat | 7.5 | 150000 | station wagon | manual | 105 | diesel | no | 5100 |
| volkswagen | passat | 10.58333 | 150000 | limousine | manual | 140 | diesel | no | 6700 |
| volkswagen | golf | 5.833333 | 50000 | station wagon | manual | 80 | gasoline | no | 9899 |
| volkswagen | polo | 12.08333 | 125000 | limousine | automatic | 75 | gasoline | no | 4900 |
| volkswagen | polo | 15.83333 | 150000 | station wagon | manual | 75 | gasoline | no | 1690 |
| volkswagen | golf | 7.666667 | 150000 | limousine | automatic | 105 | diesel | no | 8499 |
| volkswagen | polo | 11.91667 | 125000 | compact cars | manual | 72 | gasoline | no | 3000 |
| volkswagen | sharan | 10 | 150000 | bus | manual | 140 | diesel | no | 6350 |
| volkswagen | golf | 2.833333 | 40000 | convertible | manual | 211 | gasoline | no | 20450 |
| volkswagen | golf | 4.416667 | 125000 | limousine | automatic | 170 | diesel | no | 15900 |

3. Our Dataset



4. Data Preprocessing Recap

- Outliers (Too big or small value)
- Ridiculous Values
- Duplicated Data

| | | | | | | | |
|---|------|--|--|---|----------|--------|---------------|
| *_Mercedes_Benz_W220_S_400_CDI_Langvers._<>_162_TKM_>_1 | 9999 | | | 0 | s_klasse | 150000 | mercedes_benz |
| *_Mercedes_Benz_W220_S_400_CDI_Langvers._<>_162_TKM_>_1 | 9999 | | | 0 | s_klasse | 150000 | mercedes_benz |
| *_Mercedes_Benz_W220_S_400_CDI_Langvers._<>_162_TKM_>_1 | 9999 | | | 0 | s_klasse | 150000 | mercedes_benz |
| *_Mercedes_Benz_W220_S_400_CDI_Langvers._<>_162_TKM_>_1 | 9999 | | | 0 | s_klasse | 150000 | mercedes_benz |
| *_Mercedes_Benz_W220_S_400_CDI_Langvers._<>_162_TKM_>_1 | 9999 | | | 0 | s_klasse | 150000 | mercedes_benz |
| *_Mercedes_Benz_W220_S_400_CDI_Langvers._<>_162_TKM_>_1 | 9999 | | | 0 | s_klasse | 150000 | mercedes_benz |
| *_Mercedes_Benz_W220_S_400_CDI_Langvers._<>_162_TKM_>_1 | 9999 | | | 0 | s_klasse | 150000 | mercedes_benz |

| # | R... ↓ | count ↓ Number (integer) |
|-----|--------|-----------------------------|
| 155 | 9999 | 27 |
| 154 | 9996 | 1 |
| 153 | 9450 | 1 |
| 152 | 9229 | 1 |
| 151 | 9000 | 5 |
| 150 | 8888 | 2 |
| 149 | 8500 | 1 |
| 148 | 8455 | 1 |
| 147 | 8200 | 1 |
| 146 | 8000 | 2 |
| 145 | 7800 | 1 |
| 144 | 7777 | 1 |
| 143 | 7500 | 2 |
| 142 | 7100 | 1 |

| # | R... ↓ | count Number (integer) |
|----|--------|---------------------------|
| 89 | 2018 | 3 |
| 88 | 2017 | 7 |
| 87 | 2016 | 283 |
| 86 | 2015 | 2027 |
| 85 | 2014 | 3143 |
| 84 | 2013 | 3817 |
| 83 | 2012 | 5479 |
| 82 | 2011 | 6594 |
| 81 | 2010 | 6584 |
| 80 | 2009 | 7905 |
| 79 | 2008 | 8220 |
| 78 | 2007 | 8841 |
| 77 | 2006 | 9958 |
| 76 | 2005 | 9536 |

Dialog - 4:94 - Row Filter (PS를)

File

Filter Criteria | Flow Variables | Job Manager Selection | Memory Policy

Column value matching

Column to test: powerPS

filter based on collection elements

Matching criteria

use pattern matching

case sensitive match

contains wild cards

regular expression

use range checking

lower bound: 30

upper bound: 800

only missing values match

Include rows by attribute value

Exclude rows by attribute value

Include rows by number

Exclude rows by number

Include rows by RowID

Exclude rows by RowID

OK

Apply

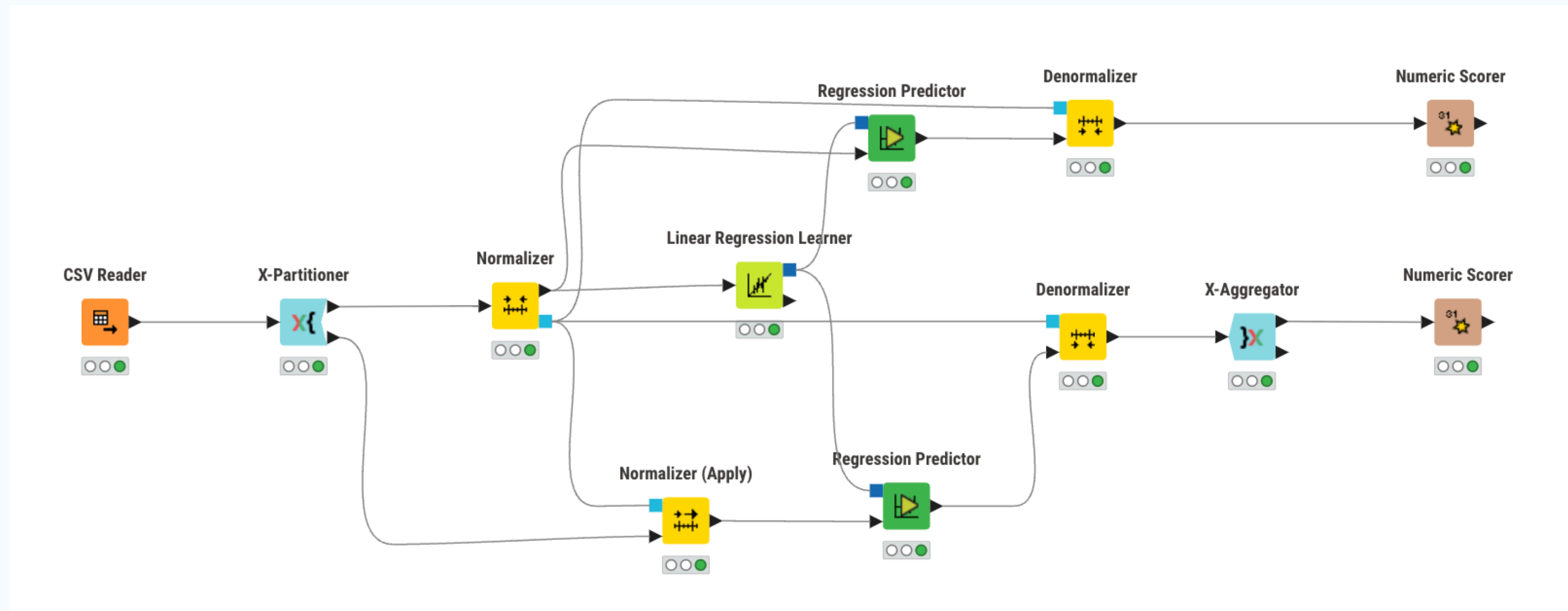
Cancel

9

5. Model - Our Evaluation metrics

| Evaluation metric | Advantage | Disadvantage | Why Selected? |
|----------------------|---|-------------------------------------|---|
| R² | Evaluate how much the model describes price volatility & explanatory | Sensitive to outliers | Suitable for assessment of explanatory power |
| MAPE | Intuitive with prediction error as a percentage | Small prices may distort the result | Intuitively assess actual predictive performance |
| RMSE | More accurate reflection of prediction errors in high-priced vehicles | Sensitive to outliers | Useful for performance evaluation of high-priced vehicle predictions |

5. Model Works - Multiple Linear Regression (MLR)



5. Model Works - Multiple Linear Regression (MLR)

Nominal values for MLR : Label encoding

Q. Why not One-hot Encoding?

A. Experienced the **Curse of Dimensionality**...🙄

Fixed) Linear correlation > 0.5 : car_age, kilometer, powerPS, car_model

vehicleType(1), gearbox(2), fuelType(3), notRepairedDamage(4)

| CASE | x | 1, 3, 4 | 2, 3, 4 | 1, 2, 3, 4 |
|--------------------------------|-------------|--------------|-------------|-------------|
| R^2 | 0.63506842 | 0.648601815 | 0.656494454 | 0.655735274 |
| mean absolute error | 3174.356195 | 3132.950155 | 3118.916731 | 3124.028734 |
| mean squared error | 25281779.31 | 24440843.29 | 23588280.12 | 23578420.14 |
| root mean squared error | 5028.098976 | 4943.768127 | 4856.776721 | 4855.761541 |
| mean signed difference | -5.4771E-10 | -4.02865E-09 | 4.30216E-10 | 4.01539E-10 |
| mean absolute percentage error | 0.906901314 | 0.903172851 | 0.910689249 | 0.899603956 |
| adjusted R^2 | 0.63506842 | 0.648601815 | 0.656494454 | 0.655735274 |

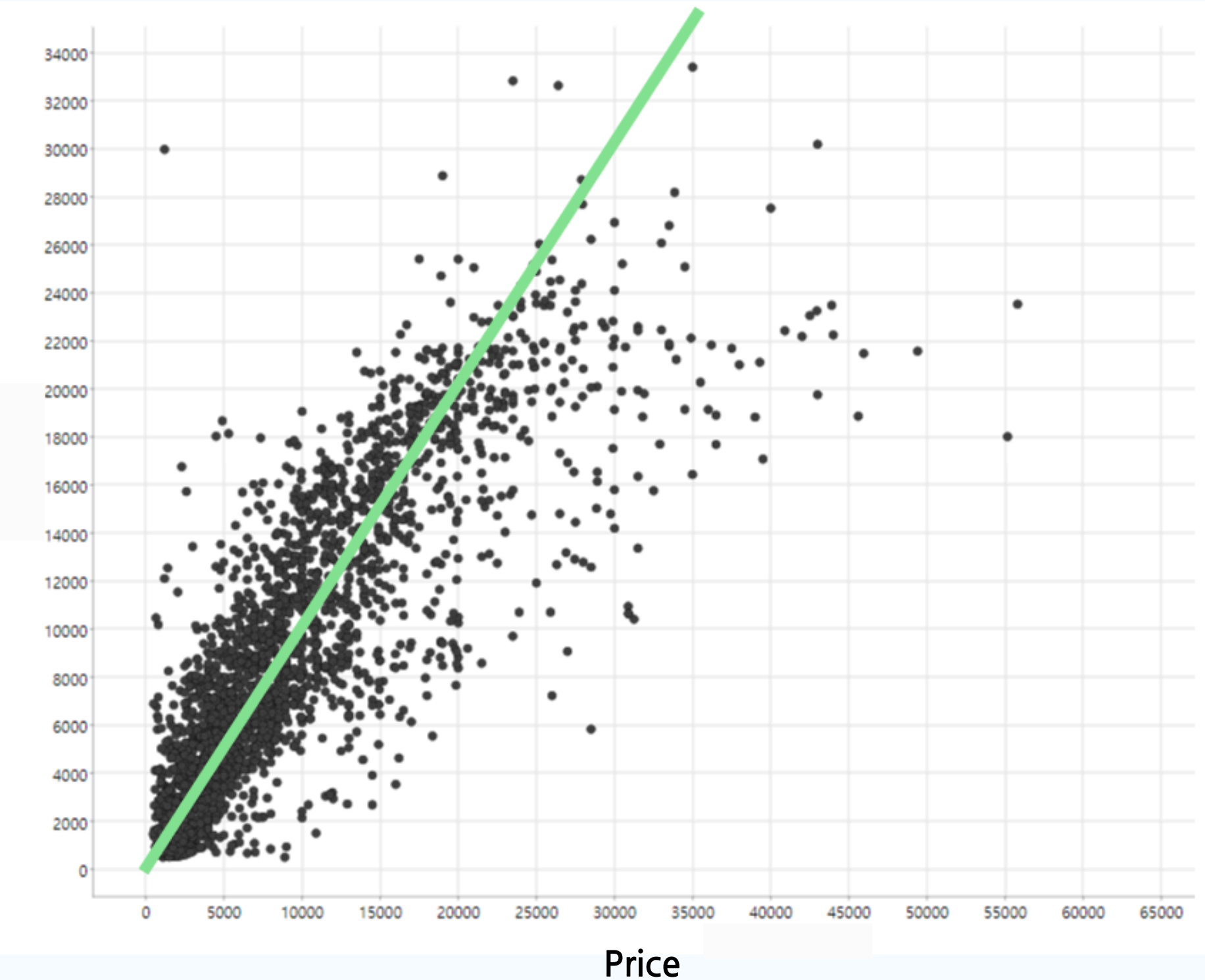
Using selected features

Using all features

5. Model Works - Multiple Linear Regression (MLR)

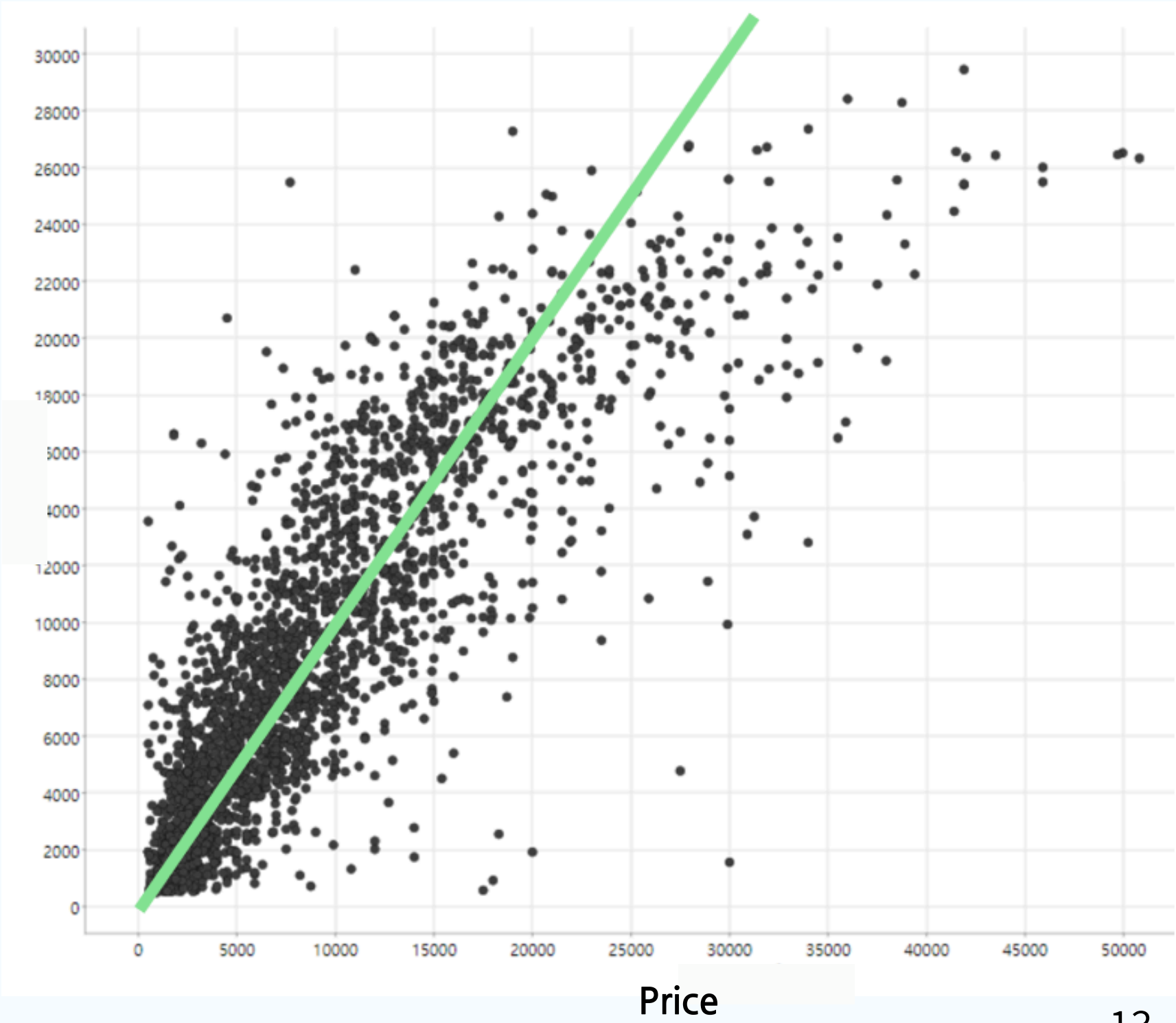
Prediction Price

Using selected features

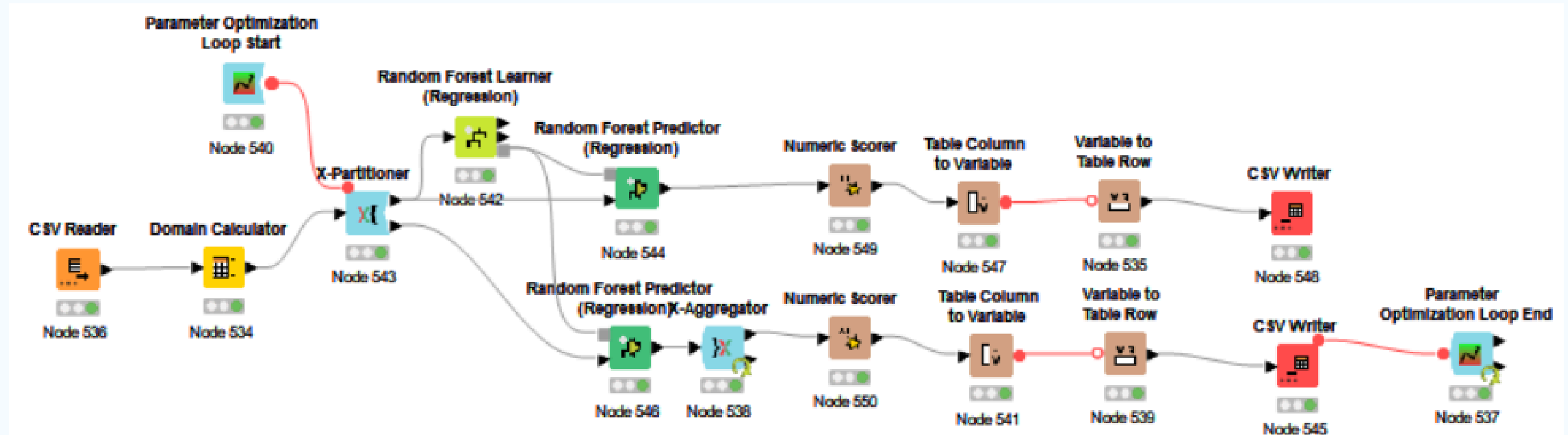


Prediction Price

✓ Using all features



5. Model Works - Random Forest



5. Model Works - Random Forest

- ★ **Limit number of levels (tree_depth)** : Number of tree levels to be learned
- ★ **Minimum child node size (min_child_size)** : Minimum number of records in child nodes

| | | | | |
|----------------|-------------|------------|-----------|------------|
| | Start Value | Stop Value | Step Size | Best Param |
| tree_depth | 3 | 100 | 5 | 73 |
| min_child_size | 5 | 30 | 5 | 5 |

| | |
|--|--------------------|
| performance | prediction (price) |
| R^2 | 0.88045 |
| MAPE (Mean Absolute Percentage Error) | 0.356 |
| RMSE (Root Mean Squared Error) | 2882.6 |

5. Model Works - Random Forest

| | Start Value | Stop Value | Step Size | Best Param |
|----------------|-------------|------------|-----------|------------|
| tree_depth | 50 | 90 | 1 | 69 |
| min_child_size | 5 | 6 | 1 | 5 |

★Optimal Hyperparameter

tree_depth : 69 / min_child_size : 5

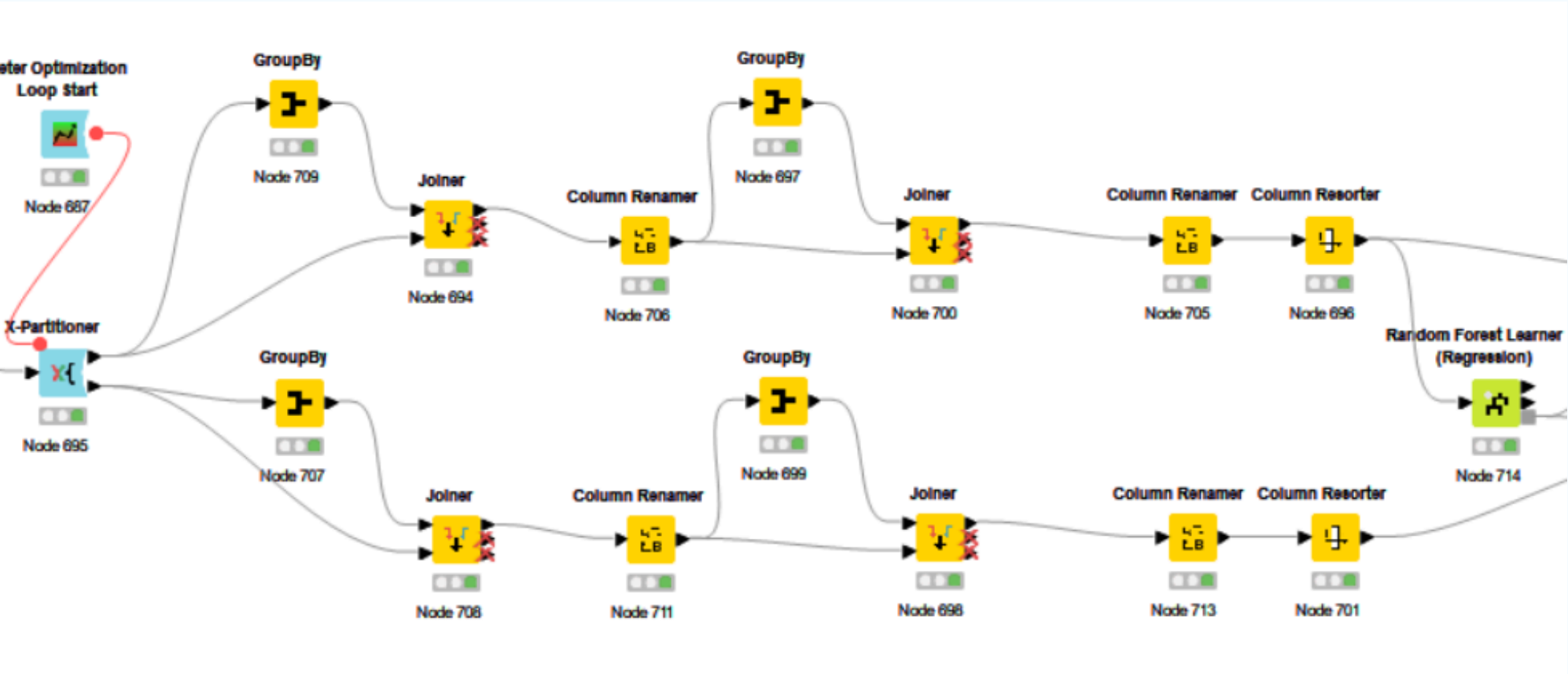
| performance | prediction (price) |
|--|--------------------|
| R^2 | 0.8808 |
| MAPE (Mean Absolute Percentage Error) | 0.356 |
| RMSE (Root Mean Squared Error) | 2878.83 |

5. Model Works - Random Forest

★Target Encoding..?

Convert brands and models to average prices

| car_brand | car_model | car_brand_enc | car_model_enc | price |
|------------|-----------|---------------|---------------|-------|
| volkswagen | golf | 6819.045426 | 6024.54069 | 600 |
| volkswagen | golf | 6819.045426 | 6024.54069 | 1459 |
| volkswagen | passat | 6819.045426 | 6013.214716 | 5890 |
| volkswagen | golf | 6819.045426 | 6024.54069 | 4000 |
| volkswagen | touareg | 6819.045426 | 17980.71884 | 11900 |
| volkswagen | beetle | 6819.045426 | 8735.380631 | 5800 |
| volkswagen | fox | 6819.045426 | 2997.835766 | 7800 |
| volkswagen | golf | 6819.045426 | 6024.54069 | 19200 |
| volkswagen | golf | 6819.045426 | 6024.54069 | 1200 |
| volkswagen | golf | 6819.045426 | 6024.54069 | 3799 |
| volkswagen | golf | 6819.045426 | 6024.54069 | 8990 |
| volkswagen | golf | 6819.045426 | 6024.54069 | 1750 |
| volkswagen | passat | 6819.045426 | 6013.214716 | 5100 |



in order to prevent ‘the data leakage’...

5. Model Works - Random Forest

| | Start Value | Stop Value | Step Size | Best Param |
|----------------|-------------|------------|-----------|------------|
| tree_depth | 90 | 110 | 1 | 100 |
| min_child_size | 5 | 6 | 1 | 5 |

★Optimal Hyperparameter

tree depth : 100 / min_child_size : 5

| performance | prediction (price) |
|--|--------------------|
| R^2 | 0.878 |
| MAPE (Mean Absolute Percentage Error) | 0.37 |
| RMSE (Root Mean Squared Error) | 2911.78 |

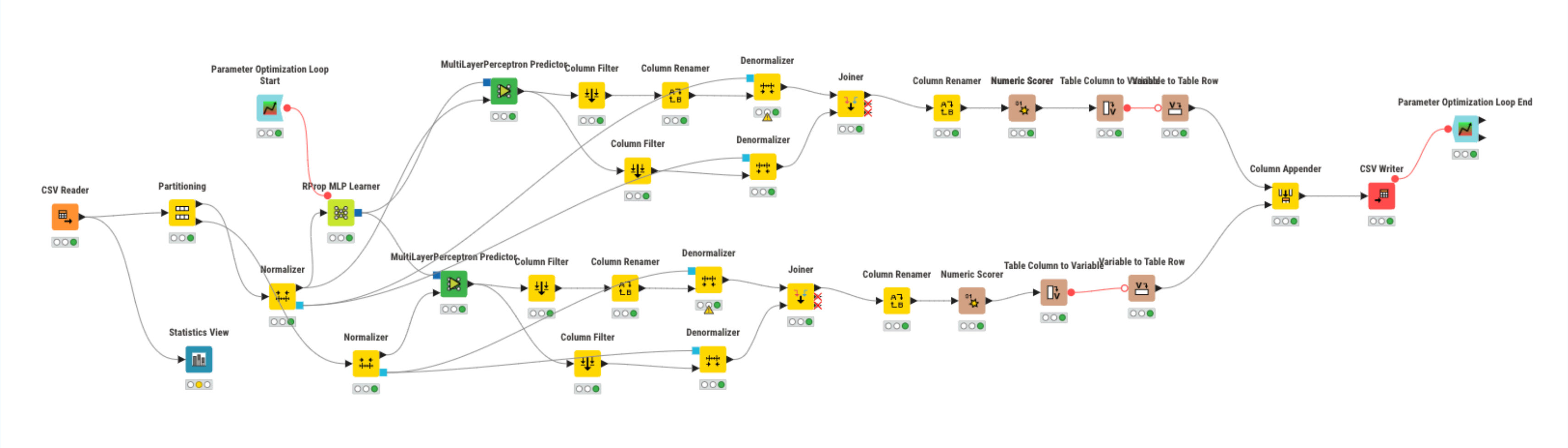
Original Dataset (Unencoded)

| | | Min child node size | | | |
|------------|----|---------------------|----------|--|--|
| | | 5 | 6 | | |
| Tree depth | 60 | 0.879586 | 0.876837 | | |
| | 61 | 0.878303 | 0.876403 | | |
| | 62 | 0.880456 | 0.876057 | | |
| | 63 | 0.878985 | 0.878191 | | |
| | 64 | 0.878234 | 0.877959 | | |
| | 65 | 0.876653 | 0.878279 | | |
| | 66 | 0.879733 | 0.878207 | | |
| | 67 | 0.878702 | 0.877476 | | |
| | 68 | 0.877936 | 0.876826 | | |
| | 69 | 0.880798 | 0.876526 | | |
| | 70 | 0.879721 | 0.876119 | | |
| | 71 | 0.87931 | 0.876935 | | |
| | 72 | 0.878709 | 0.877494 | | |
| | 73 | 0.879647 | 0.877303 | | |
| | 74 | 0.879002 | 0.875762 | | |
| | 75 | 0.878169 | 0.876279 | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |

Target Encoded Dataset

| | | Min child node size | | | |
|------------|-----|---------------------|----------|--|--|
| | | 5 | 6 | | |
| Tree depth | 90 | 0.876485 | 0.874156 | | |
| | 91 | 0.87578 | 0.875022 | | |
| | 92 | 0.877532 | 0.875158 | | |
| | 93 | 0.877572 | 0.875161 | | |
| | 94 | 0.875064 | 0.876897 | | |
| | 95 | 0.877696 | 0.87456 | | |
| | 96 | 0.877193 | 0.874161 | | |
| | 97 | 0.877141 | 0.874666 | | |
| | 98 | 0.877741 | 0.873691 | | |
| | 99 | 0.875669 | 0.874567 | | |
| | 100 | 0.878054 | 0.874959 | | |
| | 101 | 0.876756 | 0.874575 | | |
| | 102 | 0.877514 | 0.87576 | | |
| | 103 | 0.877466 | 0.874616 | | |
| | 104 | 0.877498 | 0.875535 | | |
| | 105 | 0.877946 | 0.874174 | | |
| | 106 | 0.877408 | 0.874065 | | |
| | 107 | 0.877069 | 0.876749 | | |
| | 108 | 0.877184 | 0.875546 | | |
| | 109 | 0.877206 | 0.874147 | | |
| | 110 | 0.877418 | 0.875256 | | |
| | | | | | |
| | | | | | |
| | | | | | |

5. Model Works - Multi-Layer Perceptron (MLP)



5. Model Works - Multi-Layer Perceptron (MLP)

| performance | prediction (price) |
|--|-----------------------|
| R^2 | 0.672 |
| MAPE (Mean Absolute Percentage Error) | 0.404 |
| RMSE (Root Mean Squared Error) | 4659.79 |

5. Model Works - Multi-Layer Perceptron (MLP)

Number of hidden layers

Number of hidden neurons per layer

| | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|----|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| 3 | 0.65017 | 0.615928 | 0.614016 | 0.622766 | 0.509766 | 0.573377 | 0.370194 | 0.403561 | 0.39479 |
| 4 | 0.642803 | 0.632683 | 0.6247 | 0.610568 | 0.603258 | 0.512381 | 0.557201 | 0.355451 | 0.334034 |
| 5 | 0.639394 | 0.633924 | 0.632836 | 0.629655 | 0.606527 | 0.596536 | 0.584553 | 0.548112 | 0.511847 |
| 6 | 0.643445 | 0.641206 | 0.617749 | 0.631223 | 0.621891 | 0.615441 | 0.569108 | 0.433752 | 0.361475 |
| 7 | 0.635318 | 0.639291 | 0.652641 | 0.623891 | 0.653204 | 0.605677 | 0.604397 | 0.585897 | 0.556575 |
| 8 | 0.648531 | 0.650159 | 0.642627 | 0.631599 | 0.642783 | 0.572472 | 0.640443 | 0.532577 | 0.596598 |
| 9 | 0.650508 | 0.656796 | 0.639056 | 0.64895 | 0.649066 | 0.648743 | 0.59522 | 0.576442 | 0.603677 |
| 10 | 0.669919 | 0.648926 | 0.64796 | 0.644714 | 0.639783 | 0.649168 | 0.640463 | 0.567658 | 0.588777 |
| 11 | 0.65793 | 0.65408 | 0.658042 | 0.639319 | 0.650362 | 0.651778 | 0.621998 | 0.645935 | 0.63396 |
| 12 | 0.658547 | 0.654088 | 0.644217 | 0.644421 | 0.652721 | 0.634978 | 0.638008 | 0.618387 | 0.650662 |
| 13 | 0.658306 | 0.659108 | 0.649263 | 0.654403 | 0.659177 | 0.635069 | 0.651555 | 0.642547 | 0.632144 |
| 14 | 0.661775 | 0.653396 | 0.657138 | 0.654256 | 0.657063 | 0.636648 | 0.638928 | 0.636911 | 0.638158 |
| 15 | 0.650996 | 0.651512 | 0.657177 | 0.662008 | 0.658917 | 0.646684 | 0.650571 | 0.643291 | 0.592856 |
| 16 | 0.661198 | 0.654775 | 0.656722 | 0.655743 | 0.657195 | 0.655165 | 0.639675 | 0.638021 | 0.626475 |
| 17 | 0.66177 | 0.665334 | 0.652482 | 0.655764 | 0.656715 | 0.658282 | 0.619201 | 0.650585 | 0.648881 |
| 18 | 0.668007 | 0.656835 | 0.660502 | 0.651676 | 0.651214 | 0.658392 | 0.649922 | 0.641516 | 0.636192 |
| 19 | 0.67212 | 0.666316 | 0.654828 | 0.66083 | 0.659117 | 0.655186 | 0.652515 | 0.634711 | 0.648319 |
| 20 | 0.671967 | 0.655871 | 0.663365 | 0.657645 | 0.651424 | 0.641183 | 0.647533 | 0.645191 | 0.643466 |
| 21 | 0.662363 | 0.651803 | 0.654011 | 0.647677 | 0.65368 | 0.65391 | 0.653154 | 0.649486 | 0.627727 |
| 22 | 0.66328 | 0.656507 | 0.666222 | 0.658549 | 0.657493 | 0.651311 | 0.647223 | 0.655864 | 0.645305 |
| 23 | 0.659553 | 0.669976 | 0.655335 | 0.649904 | 0.651897 | 0.643876 | 0.643862 | 0.648783 | 0.623982 |
| 24 | 0.657986 | 0.655509 | 0.658985 | 0.663448 | 0.650668 | 0.65276 | 0.64579 | 0.652146 | 0.616976 |
| 25 | 0.659514 | 0.650194 | 0.653474 | 0.664067 | 0.657976 | 0.653952 | 0.660429 | 0.651348 | 0.656586 |

6. Model Results

| Performance | Multiple Linear Regression | Random Forest | Multi-Layer Perceptron |
|--|----------------------------|---------------|------------------------|
| R^2 | 0.656 | 0.878 | 0.672 |
| MAPE (Mean Absolute Percentage Error) | 0.899 | 0.37 | 0.404 |
| RMSE (Root Mean Squared Error) | 4855.76 | 2911.78 | 4659.79 |

7. Conclusion

Q1. Why Random Forest?

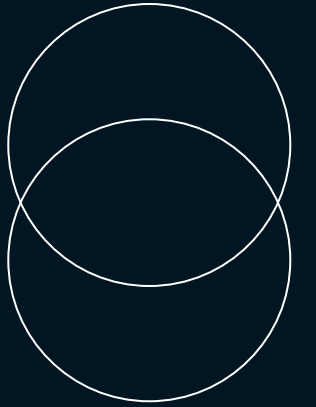
A. Since it's an ensemble model of individual decision trees...

- Good at catching **nonlinear relationships**
- Good at handling **categorical variables** well
- Can effectively reflect interactions between variables
- Automatically calculates feature importance -> **use important variables**

Q2. Why not MLP?

A. Without sufficient training data and adequate hyperparameter tuning, MLP may not capture nonlinearity well...

- Requires **learning all its characteristics** & consideration of **non-critical variables** => Can lead to a lot of unnecessary learning!



Q & A

Thank you for listening!