04. Part 2 - House prices

In this second part, you will work on the house prices dataset assembled and published by Dean De Cock. It's a set of 2,930 observations with 82 attributes each. The goal is to go through all the main steps of a data science project i.e. **preparing the data, exploring the data (EDA) and modeling**. In the modeling part, you will use the first 2,430 ones (i.e. training set) to fit and evaluate different models and use them to make predictions for the last 500 ones (i.e. test set). Note that we don't provide the prices for those 500 houses, your task is to estimate them.

A quick look at the data

Here are the first five entries from `house-prices.csv`

	Order	PID	MS SubClass	MS Zoning	...	Yr Sold	Sale Type	Sale Condition	SalePrice
0	484	528275070	60	RL	...	2009	WD	Normal	236000
1	2586	535305120	20	RL	...	2006	WD	Normal	155000
2	2289	923228250	160	RM	...	2007	WD	Normal	75000
3	142	535152150	20	RL	...	2010	WD	Normal	165500
4	2042	903475060	190	RM	...	2007	WD	Normal	122000

You can find a detailed description of each variable in the `documentation.txt` file, but there are a few things to know.

- The `Order` and `PID` variables are identifiers. They are not useful to predict house prices.
- The variables are not necessarily encoded consistently. For instance, `MS SubClass` (the type of dwelling) and `MS Zoning` (zoning classification) are both categorical variables, but one is encoded with numerical values and the other with short labels.
- The data isn't clean: there are incorrect and missing values, outliers and inconsistencies

Exploratory data analysis and data cleaning

You should gain a **comprehensive overview over the data** by **exploring and visualising** the data in various ways, including the distribution of the feature values in individual features. Throughout you should **comment** on your observation, **discuss** how they might affect later steps in the project (e.g. insights that help with feature engineering and data preprocessing for the modeling part) and state which **decisions** you take as a result.

You should use your EDA to identify issues with the data that require **data cleaning**. For instance

- Find and handle incorrect, missing values
- Correct inconsistencies in the variables
- Handle outliers

You are free to choose your preferred approach to handle each step. For instance, you might want to replace missing values with the average or the most frequent value or create a `missing` category. In any case, **justify your choices**!

Remember our overall objective is to predict the sale price. Hence it is useful to know which features might exhibit a relationship with our target, and which ones may be not? This will help us decide which features might be useful for our models, and which need additional feature engineering first. Start by **exploring and visualizing** the relationship between individual features and the target variables. **Comment your observation and discuss their impact on the modeling part.**

Don't forget to choose appropriate visualizations and analyses depending on whether we are dealing with continuous, discrete or categorical features.

Feature engineering

Your analysis should also include **feature engineering**. Here are a few ideas

- Create indicator variables ex. year of construction is older than some threshold
- Transformations ex. log-transforms, polynomials

Suggestion: write down your feature engineering ideas during the data exploration stage.

> **Warning**
>
> **Warning**: Be careful when adding total counts (ex. the total number of rooms, living surface) and other linear combinations of the input features. If you keep the original features in the data, then those variables don't add "modeling power" to the model and can lead to ill-conditioning and numerical issues. On the other hand, if you create such variables and remove the original features, it can be seen as a way to compress the information on fewer dimensions which can be useful for the simple and intermediate models where the number of variables is limited.

Feature encoding

Your analysis should include the necessary **feature encoding steps**. The `documentation.txt` file labels each variable with its type. For categorical ones, it uses the ordinal and nominal classification.

- Ordinal variables - you can order the categories
- Nominal variables - no possible ordering of the categories

The encoding depends on the type of variable and its meaning. For instance, the kitchen quality variable is on a scale from excellent to poor. Hence, it's an ordinal variable, and you can choose to apply one-hot encoding or define a numerical scale ex. excellent corresponds to 5 and poor to 1. In any case, **justify your choices**!

Splitting data

You should split the data into training and validation sets (e.g. 60-40 split). You will use the training set for fitting the models and the validation set for evaluating the models and tuning hyperparameters.

Model fitting

Your analysis should include an appropriate **baseline** and evaluate three different models ranging in complexity

- A **baseline** that entails no modeling, and supposedly should be beaten by the three

models
- A **simple model** with two variables (three with the target variable)
- An **intermediate model** (between 10 and 20 variables)
- A **complex model** with all variables

The number of variables is only given as an indication, it's not a strict range. Also, it corresponds to the variables count before one-hot encoding. For the simple and intermediate models, you can choose the variables. You are free to choose your preferred approach for this variable selection step, but you should include a short comment to **explain your choice**.

> **Note**
>
> **Example**: I decide to choose variables `v1` , `v2` and `v3` for my simple model because I think that they provide a good overview of the house - or - I choose these variables because they are the most correlated with the target - or - I decide to test the SelectKBest object that I found in Scikit-learn to do automatic feature selection.

Evaluation metrics

You can also track different metrics to evaluate the performance of your models. However, make sure to print the **mean absolute error** (MAE) score **in dollars** for each one.

> **Note**
>
> **Example**: My simple model has an MAE of 25,123 thousand dollars - or - The MAE score of my model is `.. ± ..` dollars (MAE ± std)

Regularization

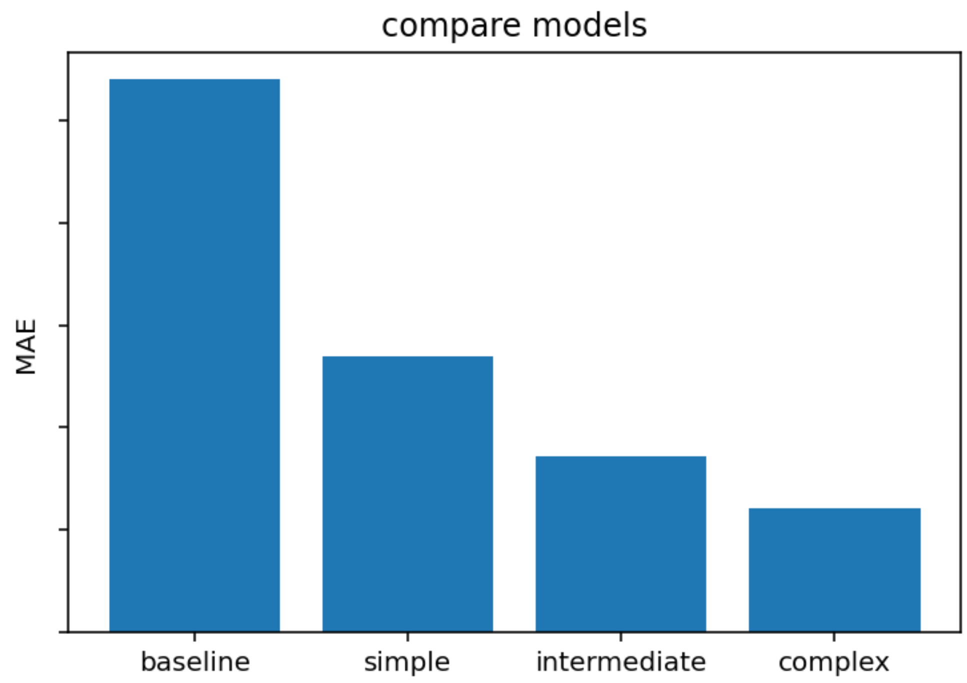
Your analysis should include **regularization** for the complex model.

- Briefly explain the **objective** of regularization, and how it will make the complex model different from other models
- Tune regularization strength with **grid search**
- Plot the **training and validation curves**
- Discuss what you observe in the plot, e.g. potential **overfitting**

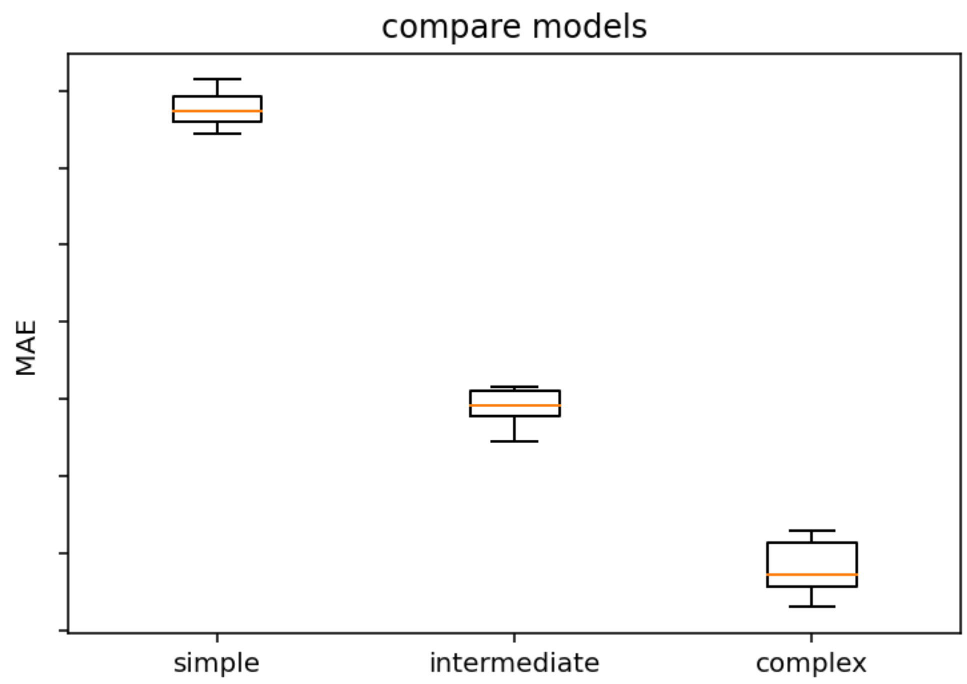
Communicating the results

You are free to use any appropriate cost function to fit your models, with or without prices transformation ex. log-transformed. However, **explain your choices** in the notebook.

Your analysis should also include a final visualization which summarizes the different models MAE scores. For instance, using a bar chart



or even a box-plot if you decided to evaluate your models on several train/test splits



Predicting on test data

Now that you have your three models ready, let's go to the 500 new houses that are unseen by your models. They make your test set. Get the attributes of these houses from `house-prices-test.csv` and predict their prices using each model and save them in a `.csv` file.

- Predictions from your simple model – `predictions-simple-model.csv`
- Predictions from your intermediate model – `predictions-intermediate-model.csv`
- Predictions from your complex model – `predictions-complex-model.csv`

Your `.csv` files must contain 500 rows and 2 columns: the house **`PID`** and the predicted price as **`SalePrice`**. You can find a sample submission file in `predictions-example.csv`. Be careful to respect the column names.

	PID	SalePrice
0	909279080	0
1	907126050	0
2	528144030	0
3	535452060	0
4	911202100	0

Evaluate your predictions

Please check your predictions sanity and performances by yourself using exts-p3review.herokuapp.com.
Just upload the three `.csv` files there and if filenames and formats are valid, you will be

printed the test MAEs!

QUESTIONS

Ask a Question

miguelfaf · Learner · 8 days ago

When to apply feature encoding

1 Answer

It seems to me that feature encoding with a numerical scale, of ordinal variables, should be done before feature engineering. This way it could inform the feature engineering step, by e.g. computing correlations. Does this make sense, or should we stick to the EDA steps order provided in the task description?

POST ANSWER

Amir Khalilzadeh · Teacher · 4 days ago

0

In general this is an iterative process where we go back and forth between steps. Encoding can be part of feature engineering. So feel free to shuffle the steps so that they form the big picture (understanding the data and preparing it for modeling) in a meaningful way.

tkl · Learner · 18 days ago

Varying regressor objects between models

1 Answer

Can we use varying types of regression models between the simple, intermediate, and complex models ? E.g. Linear Regressions for the first two and SGDRegressor for the last one ?

I understand this may impact the comparability of the MAE score, but a linear regression may make more sense with a reduced number of parameters, while for the complex model a regression method allowing for regularization and hyperparameter tuning will be more appropriate.

POST ANSWER

Amir Khalilzadeh · Teacher · 17 days ago

1

Yes, it makes sense to use Linear Regression for simple and intermediate models. But for the complex model I suggest Ridge regression. The SGDRegressor suits situations where sample size is super large. The simple structure of Ridge regression allows to understand the necessity, implementation and impact of regularization. It will make it easier to digest similar technics in more complex models you will learn in the next course.

limegreen-mandarine · Learner · 2 months ago

I have split the data frame and treated missing values in the test data set in the same way as in the training data set. Should I also treat inconsistencies or outliers in a test data set as I will for the training data set?

1 Answer

POST ANSWER

Amir Khalilzadeh · Teacher · a month ago

0

There are two things to consider when treating test data:

- The number of houses in the test set should remain the same after addressing the issues in the test set.

- You should bring the necessary information from the train data to fix issues in the test data (eg fill missing). This is because the test data is not aimed for 'learning' but just for the final evaluation.

In general, the data in the test set should be consistent with the train data and the problem at hand. For instance, we should drop a commercial building from the test data if the objective is to build a model that predicts house prices. Similarly, a model aimed to predict the temperature in Switzerland shouldn't be used to predict temperature in a nordic country. So, you can check and fix inconsistencies or outliers in a test data but to consider what is consistent or in-liner you should be relying on the train data. For this project you can skip this step or do only some general consistency checks.

Tiago · Learner · a year ago

Complex model variables

✓ 1
Answer

Hi,

We already know that some variables are not useful for prediction, such as PID, and therefore should not be included in the data we give to the model. However, there are a few other variables that also do not provide any information, yet it is specified that the complex model should contain "all" variables. Does this mean all columns except identifiers and target or can we drop a few other features even for the complex model? How about combining variables (ex. total nb rooms), can this be used for the complex model instead of the individual nb rooms features or only for simple and intermediate models?

Thanks!

POST ANSWER

ChristianLuebbe · Teacher · a year ago

✓ 0

No if you have reason/evidence to exclude variables because they don't provide sufficient information then you can of course exclude them.
The term "all" variables tries to distinguish the complex model from the other two models where we actively select a subset of features.

Tiago · Learner · a year ago

Separate file for utilities functions

✓ 1
Answer

Hello,

Is it ok to add a separate file with tools and utilities functions (coded by us of course), that is submitted as part of the project and imported in the main solution notebook file? This is to avoid having a big section of functions in the middle of the notebook and keep the code clean.

Thanks,
Tiago

POST ANSWER

ChristianLuebbe · Teacher · a year ago

✓ 0

Yes that is fine
But please ensure that the functions are clearly documented in such a file as well as in the workflow of the main notebook.

rosybrown-plum · Learner · a year ago

final predictions on new data

✓ 2
Answers

Hello,

I have a question about the final predictions on the houes-prices-test.csv data after I have trained and evaluated my 3 models. Do I have to treat the new data before feeding them into the models for predictions? For example, do I have to treat the new data for outliers, inconsistencies and missing values? and handle the numerical and categorical variables, one-hot encoding etc?
Or do I just submit the new data without preprocessing?

Thank you!

POST ANSWER

rosybrown-plum · Learner · a year ago

✓ 0

My idea would be to fix all the missing values, engineer the same features and do the same encoding of categorical variables, so to have my dataset in the same frame. I would not however address the outliers. would that be correct or should I address the outliers too?

Amir Khalilzadeh · Teacher · a year ago

✓ 1

Yes, your idea is correct. It is important to preprocess the new data before feeding it into your trained models for predictions. This preprocessing step typically includes handling inconsistencies, and missing values, as well as encoding categorical variables (such as one-hot or integer encoding). By treating the new data in the same way as your training data, you ensure that the predictions from your models are accurate and meaningful.

darkviolet-raspberry · Learner · a year ago

train-test mismatch in columns after OHE

✓ 1
Answer

Hello,

After OHE some columns are omitted in test because the category is not present, so the column is not created. This leads to a mismatch between train and test. Any advice on this?

Thank you

POST ANSWER

Amir Khalilzadeh · Teacher · a year ago

✓ 2

Hi, encountering a mismatch between the number of columns in the training and test datasets after one-hot encoding (OHE) is a common issue. This occurs when certain categories present in the training set are not present in the test set.

To handle this issue, it is important to ensure that the same set of categories is present in both the training and test datasets. Here are a few potential solutions:

- Add missing categories: If a category is missing in the test set but present in the training set, you can manually add a column with zeros for that category in the test set. This will ensure consistency in the number of columns.
- Drop inconsistent columns: If a category is missing in the training set but

present in the test set, you can drop the corresponding column from the test set. This ensures consistency in the number of columns, but keep in mind that you are losing information in this case.

To quickly solve this you can use the `reindex` function as explained here, which basically creates placeholders for missing columns.

Hope this helps!

darkviolet-raspberry · Learner · a year ago

MAE in dollars when the target is log-transformed

1 Answer

Hello,

```
...make sure to print the mean absolute error (MAE) score in dollars...
```

If I log-transform the target before fitting, how do I "un-log" to show MAE in dollars?

Thank you.

POST ANSWER

ChristianLuebbe · Teacher · a year ago

1

If you log-transformed the target then your model makes predictions in that scale. To calculate the MAE in dollars you first convert the log-scale predictions back to the dollar scale using the exponential function. Afterwards you calculate the MAE in the usual way.

rosybrown-plum · Learner · a year ago

handling the outliers

2 Answers

Hi,
I am trying to address the outliers in the order to prepare the data. However, once again I am not sure to which extent I should clean the data. I have addressed the missing values and some inconsistencies, but when I apply z-score or IQR for detection of outliers, a lot of data appear as outliers that are just far away from the mean, but not necessarily wrong. For example the houses with very large Porch areas...So I am not sure if this is something that should be removed or not, as it is not part of inconsistencies but simply far from the average value. Could you please clarify more on the extent of outliers removal.
Thank you

POST ANSWER

Amir Khalilzadeh · Teacher · a year ago

0

Handling outliers is an important step in data preparation. However, it is important to understand the context of the data and the nature of the outliers before deciding whether to remove them or not. You made two key points: **a lot of data appear as outliers** but **not necessarily wrong**. Outliers that are just far away from the mean, but not necessarily wrong, should be examined closely to determine if they are valid data points. In the example you provided, houses with very large porch areas may be valid data points and should not be removed simply because they are far from the average value. You can also inspect *other features* of suspicious houses to make a decision. For instance, a house with 10 rooms may show up on your radar as suspicious, but you can keep it if its Lot Area is convincing.

Ultimately, the extent of outliers removal depends on the size of data, the specific context of the data and the goals of the analysis. It is important to use domain knowledge and common sense when deciding whether to remove outliers. And to document the steps taken to handle outliers and justify any decisions made regarding outlier removal.

rosybrown-plum · Learner · a year ago

Thank you very much for such quick response! All clear now :)

1

PIWeb · Learner · 2 years ago

I`m trying the use SelectKBest and I'm running into a warning.

1 Answer

Hi,

I`m trying the use `SelectKBest` and I'm running into a warning.

`RuntimeWarning: invalid value encountered in true_divide correlation_coefficient /= X_norms`

```
# Instantiate the SelectKBest object and
# specify the number of features you want to select and the scoring
function:
from sklearn.feature_selection import SelectKBest, f_regression

selector_v1 = SelectKBest(score_func=f_regression, k=2).fit_transform(X_tr,
y_tr)
selector_v1.shape
```

It's trying to calculate the correlation coefficient and divides it by the feature norms, and if a feature has a norm of zero, this division will raise a **RuntimeWarning**

How should I deal with this **RuntimeWarning**

- Should I remove zero?
- Should I not fill the NaN values by zero?
- When creating indicators variable and setting the lower than a threshold to zero, is it appropriate to use zero?

POST ANSWER

Amir Khalilzadeh · Teacher · 2 years ago

Hi, thanks for asking this question.
You probably have a column in `X_tr` that holds a constant value e.g. only 1s or only 0s coming from one-hot encoding. Find it and remove it.
The `X_norms` holds the standard deviation which is zero for a constant vector.
hope this helps

0

green-mulberry · Learner · 2 years ago

Addressing missing values in numerical columns

1 Answer

Hello,

what is the best way to handle missing values in numerical columns (for example Garage year built) which actually stand for no garage, rather than missing data ?

POST ANSWER

ChristianLuebbe · Teacher · 2 years ago

As so often we have to decide this on a case-by-case basis.
If the existing data is not very informative enough then we might drop the feature altogether. If we want to keep the feature then we definitely need to

1

provide a numerical value to the model.
We might ask ourselves how that feature might generally impact the target value. Based on that we would decide which value per garage might be most suitable/least damaging. So we could ask: When would the sale price be higher, respectively lower? Are there other values in the data that behave in a similar way and could act as a suitable proxy?

Shady · Learner · 3 years ago

How to handle ordinal encoding for a large set of ordinal variables/columns?

✓ 1 Answer

For the Feature encoding section, I ideally would like to assign a numeric scale for the ordinal variables through Ordinal Encoding.

However, there are 24 ordinal variables. In such a case, how realistic would it be to go through each one and assign a numeric scale for its inputs?

In this case, I guess one-hot encoding is my only option for ordinal variables if I don't want to go through each one of them? or is there any other type of encoding that could apply?

Thank you

POST ANSWER

Michael Notter · Teacher · 3 years ago

✓ 1

Thank you for your question. It probably depends on which stage of the project you are. At the very beginning, when you want to get a quick "lay of the land" using a 'one-fits-all' approach (e.g. using ordinal feature encoding or one-hot encoding) might lead to some reasonable results. So, if the goal is to quickly establish a full ML-pipeline, from start to end, this might serve your purpose.

However, if you want to get serious with your data preparation, EDA and modelling part, then you should always make sure that you understand the encoding of your features and chose an appropriate approach. One-hot encoding might be a quick solution, but it can create a lot of new features, plus it can obscure relationships to either features if you don't do a detailed enough EDA. But if your ordinal variable is not nicely linearly ordered, than applying a quick ordinal encoding will also not work.

So in short: "If you don't want to go through each one of them", then you need to accept that your data preparation is most likely not finished or good enough and your end results might not be good (enough).

Also, keep in mind that 80% of any data science project usually is spent on data preparation, data cleaning, feature encoding, and other EDA steps. Only 20% is about creating and fine-tuning model and results investigation.