MGMT 675

AI-ASSISTED FINANCIAL ANALYSIS

RICE | BUSINESS
Jones Graduate School of Business

INTRODUCTION TO MACHINE LEARNING

OBJECTIVE OF MACHINE LEARNING

- Goal is prediction (or making optimal decisions in reinforcement learning)
- Linear regression is a machine learning model (usually not the best) but we don’t do t-stats or p-values in machine learning.
- Only question is how good are the predictions on data the machine was not trained on - can we trust the predictions on new data?

LEARN = TRAIN = FIT = ESTIMATE

- Estimating a model like linear regression is called “fitting the model”or “training the model.”
- To say a machine “learns” from data means its parameters are estimated from the data.
- The predictors (x variables) are called features.

REGRESSION VS CLASSIFICATION

- Regression means to predict a continuous variable (not necessarily linear regression).
- Classification is to predict a categorical variable. Binary or multiclass.

MACHINE LEARNING IN FINANCE

- Fraud detection
- Credit risk analysis
- Return prediction
- Valuation
- Sentiment analysis
- Time series forecasting

TRAIN AND TEST

- To assess how well a model will perform on new data, we hold out some data when training it.
- Split our data (usually randomly) into train and test subsets.
- For return prediction etc., we may split on some date and train on older data and test on newer data.
- Train on the training data and test on the test data.

ASSESSING PERFORMANCE

- How do we decide if performance is good or bad?
- For continuous variables,
  - usually want to achieve a low sum of squared errors
  - equivalently, achieve a high R^2.

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2}$$

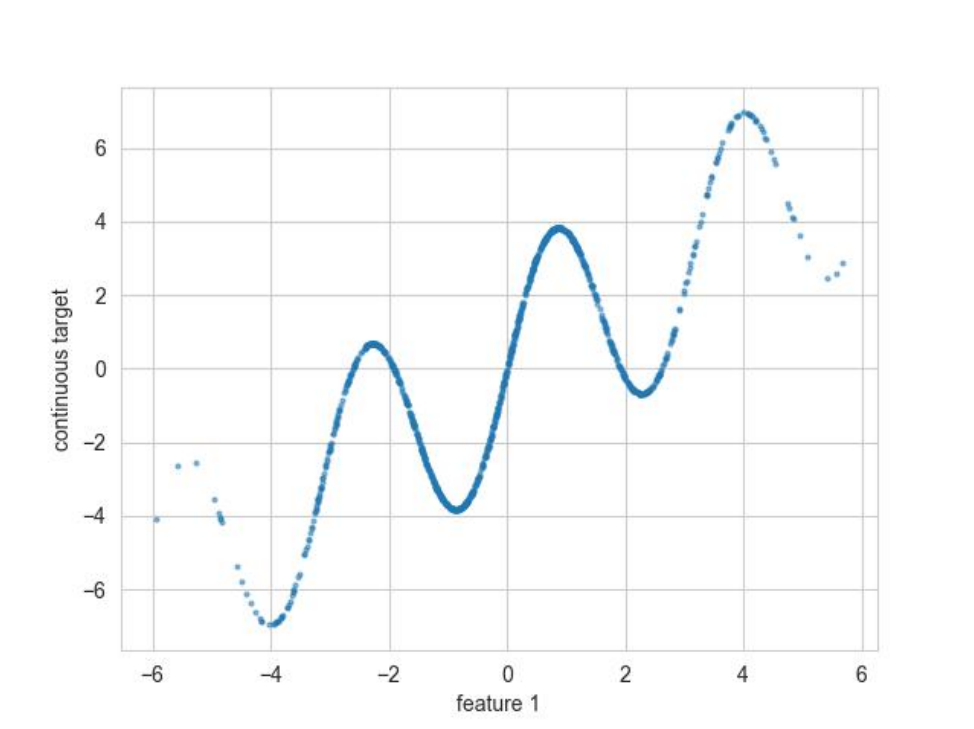
- For categorical, can use % accurately classified.

EXAMPLE

- Download mldata.xlsx from the course website and upload to Julius.
- Tell Julius you want to predict the “continuous” variable using x1 through x100 as the features.
- Ask Julius to recommend a model.
- Ask Julius to fit the model and report the scores on the training data and the test data.

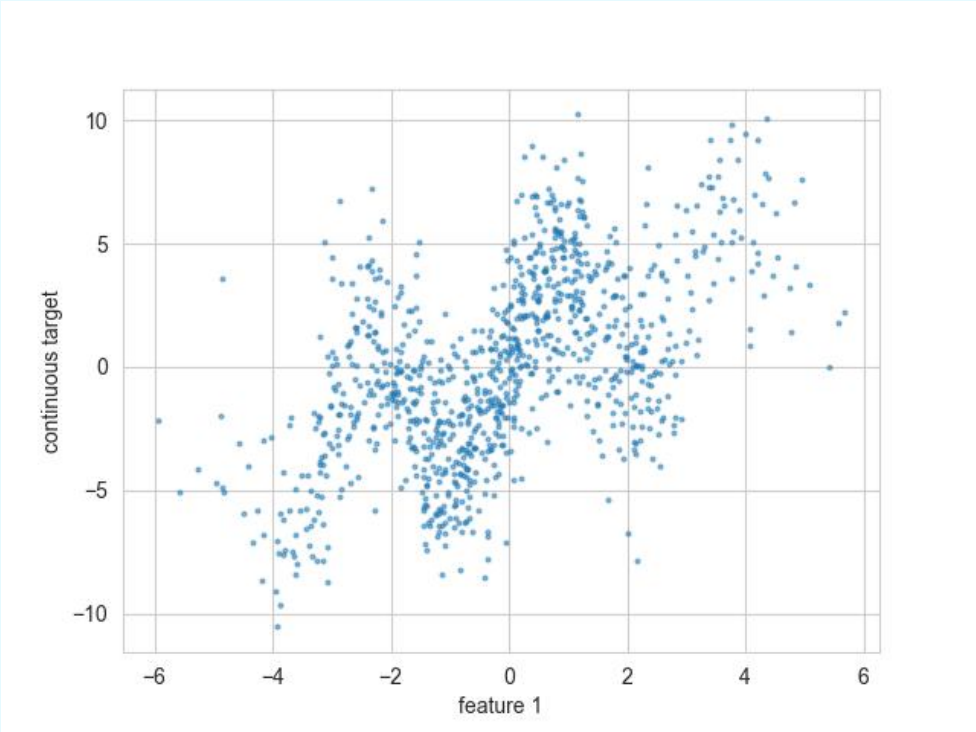
HOW WAS THE DATA GENERATED?

- This is the true model.
- x1 is the only useful feature.



DATA WE'RE FEEDING THE MODEL

- Added noise
- And added 99 other completely irrelevant features



OVERFITTING AND SHRINKAGE

UNDERFITTING AND OVERFITTING

- A model is underfit if it makes poor predictions on its training data.
- A model is overfit if it makes good predictions on its training data but performs poorly on new data.
- Overfitting means that the chosen parameters reflect chance relationships in the training data.

MODEL COMPLEXITY AND SHRINKAGE

- We can create more complex models by increasing the number of parameters (like adding variables in a linear regression).
- We can reduce complexity by reducing the number of parameters. Or by choosing less influential parameter values (like regression coefficients closer to zero).
- A more complex model is less likely to underfit but more likely to overfit.

SHRINKAGE/REGULARIZATION/PENALIZATION

- To induce selection of parameter values that are less influential, we can penalize large values.
- Example: lasso and ridge regression
  - Usually in linear regression, we minimize SSE (sum of squared errors).
  - LASSO: minimize SSE + penalty $\times$ sum of $|\beta_i|$.
  - Ridge: minimize SSE + penalty $\times$ sum of β_i^2.

EXAMPLE

Ask Julius to vary the penalty parameter (called alpha) in ridge regression and to report the scores on the training and test data.

VALIDATING HYPERPARAMETERS

HYPERPARAMETERS

- The penalty parameter in lasso or ridge is called a hyperparameter.
- Hyperparameters are model parameters that are not fit from the data during training.
- Instead they are specified before training.
- How to choose the best hyperparameter values?

DON'T OPTIMIZE ON YOUR TEST DATA

- Choosing the best hyperparameter using the test data may overfit the test data.
- The test data may no longer provide a reliable test of how the model will perform on new data.
- Instead, we could hold out some data within the training data (called validation data) for doing model comparison.

CROSS VALIDATION

- Instead of splitting the training data once into train/validate, we can do it multiple times.
- This is called cross validation

- Split the training data into random subsets, say, A, B, C, D, and E.
- Use $A \cup B \cup C \cup D$ as training data and validate on E.
- Then use $B \cup C \cup D \cup E$ as training data and validate on A.
- Then, …, until we have trained and validated 5 times (or as many times as we want).

- For each train/validate split,
  - Train for multiple values of the hyperparameter.
  - Compute the score for each on the validation data.
- This produces 5 scores for each value of the hyperparameter. Average them.
- Choose the value that produces the highest average score.
- Then proceed to testing on the test data.

GRID SEARCH OR RANDOMIZED SEARCH

- We can specify the values of the hyperparameters to consider - called grid search.
  - For example, alpha in [0.01, 0.1, 1, 10]
- Or we can use randomized search over a range of hyperparamer values.
  - For example, alpha in [0.01, 10]

SCALING/TRANSFORMING FEATURES

WHY SCALE OR TRANSFORM?

- A coefficient of 2 on a variable that ranges from 1,000 to 10,000 is very different from a coefficient of 2 on a variable that ranges from 0.1 to 1.0.
- We should penalize the former more aggressively.
- Or, better, rescale the variables so they are on the same scale before training a model.

STANDARD SCALER/QUANTILE TRANSFORMER

- Scikit-learn’s standard scaler will subtract the mean and divide by standard deviation, so each feature has a zero mean and standard deviation = 1.
- Quantile transformer can be better for outliers.
  - Can transform to normal(0, 1) or uniform on (0, 1)

REMEMBERING TRANSFORMATIONS

- Suppose you scale or transform features and train a model.
- Now you get a new observation and need to make a prediction. How do you scale or transform the new observation?

- You need to apply the same transformation you applied to the training data. So, your model needs to remember it.

PIPELINES

- Easiest way is to put scale/transform and model fit in a pipeline.
- Then fit the pipeline. Use the pipeline to predict.
- Can also run cross validation on the pipeline.