Microsoft

MLADS

MACHINE LEARNING, AI,
AND DATA SCIENCE CONFERENCE

November 14- 18, 2022

1

Microsoft

Introduction to Machine Learning Algorithms

Shep Sheppard
shep.sheppard@microsoft.com
https://www.linkedin.com/in/sqlshep/
@sqlshep

2

Microsoft

Introduction to Data Engineering

Shep Sheppard
Customer Engineer
Fast Track ISV

3

Session goals

- Introduce you to the common ML Algorithms ant topics
  - Supervised Learning
  - Unsupervised Learning
  - Regression
  - Classification
  - Clustering
  - Scoring a Regression Model
  - Scoring a Classification Model

4

4

Agenda

Supervised Learning
Unsupervised Learning
What is Regression
What is Classification
What Machine Learning Models do I use
How do I know if a Model is any good

5

5

Unsupervised Learning

- In unsupervised learning, the data points aren't labeled
- You are asking the model to find the patterns and label the data for you
- This can be used for tasks like;
  - Clustering
  - Topic Modeling
  - Embeddings
  - Anomaly Detection
  - Dimensionality Reduction
  -

06- K-means Clustering

6

6

Supervised Learning

- Algorithms make predictions based on a set of labeled examples that you provide
- Classification
- Regression
- Recommendation Systems

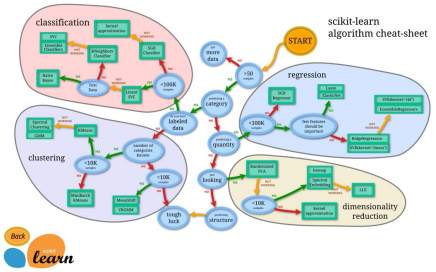
7

7

Model Selection - Type Business problem

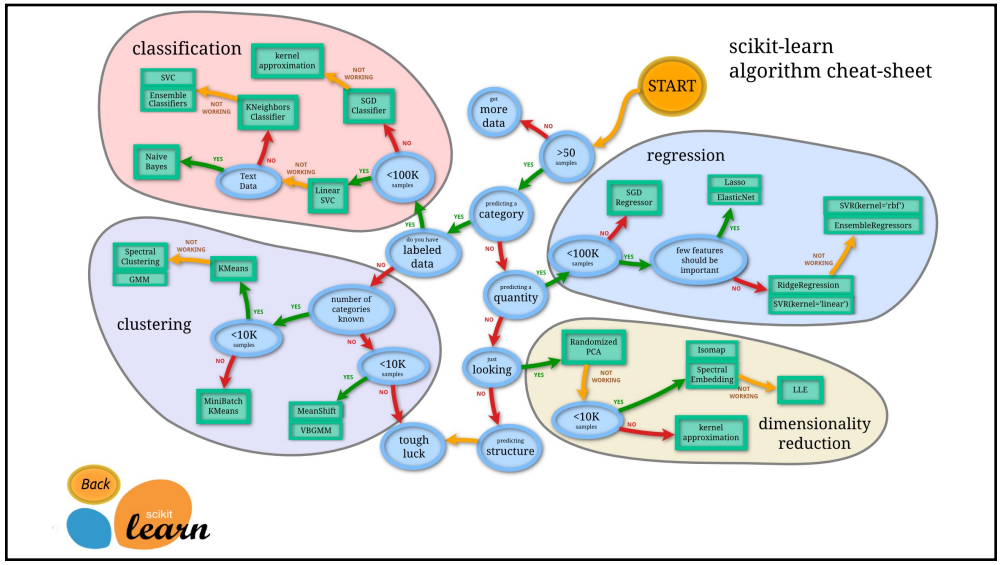
- The goal of model selection and training is to answer the business problem indicated in step one of the ML process

- Regression
- Two-Classification
- Multiclass Classification
- Text Analytics
- Clustering
- Recommenders
- Image Classifications

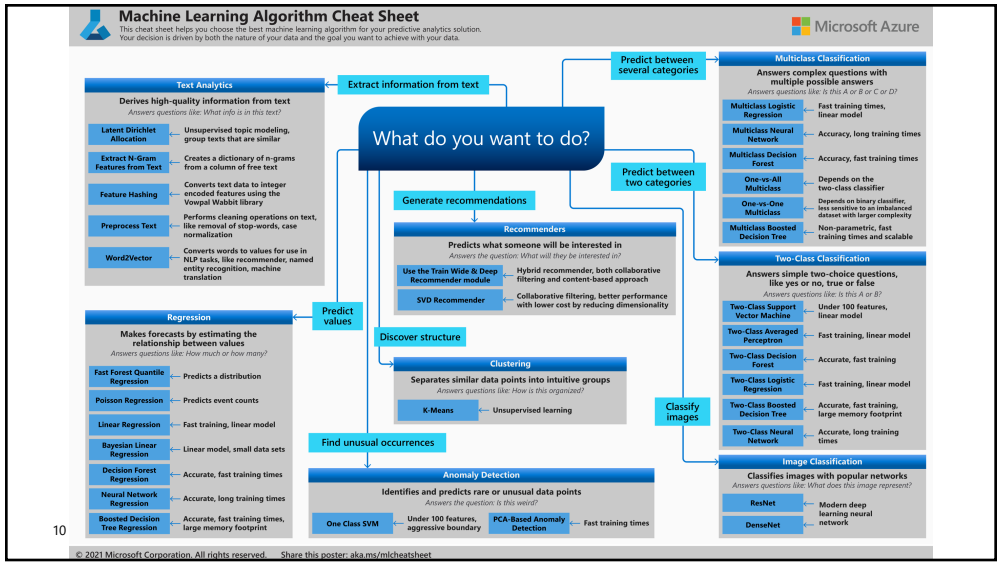


8

8


scikit-learn
algorithm cheat-sheet
START
classification
kernel approximation
SVC
Ensemble Classifiers
KNeighbors Classifier
SGD Classifier
Naive Bayes
Text Data
Linear SVC
<100K samples
NOT WORKING
YES
NO
regression
SGD Regressor
Lasso
ElasticNet
SVR(kernel='rbf')
EnsembleRegressors
RidgeRegression
SVR(kernel='linear')
few features should be important
<100K samples
>50 samples
got more data
predicting a category
do you have labeled data
predicting a quantity
just looking
predicting structure
tough luck
clustering
Spectral Clustering
GMM
KMeans
number of categories known
<10K samples
MiniBatch KMeans
MeanShift
VBGMM
dimensionality reduction
Randomized PCA
Isomap
Spectral Embedding
LLE
kernel approximation
Back
scikit learn

9

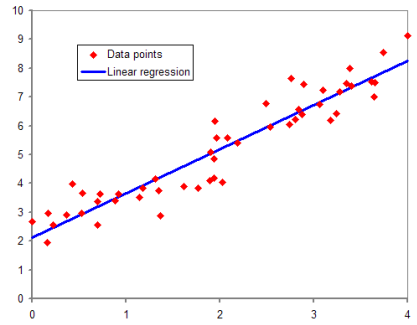

Machine Learning Algorithm Cheat Sheet
This cheat sheet helps you choose the best machine learning algorithm for your predictive analytics solution.
Your decision is driven by both the nature of your data and the goal you want to achieve with your data.
Microsoft Azure
What do you want to do?
Extract information from text
Predict between several categories
Predict between two categories
Generate recommendations
Predict values
Discover structure
Find unusual occurrences
Classify images
Text Analytics
Derives high-quality information from text
Answers questions like: What info is in this text?
Latent Dirichlet Allocation
Unsupervised topic modeling, group texts that are similar
Extract N-Gram Features from Text
Creates a dictionary of n-grams from a column of free text
Feature Hashing
Converts text data to integer encoded features using the Vowpal Wabbit library
Preprocess Text
Performs cleaning operations on text, like removal of stop-words, case normalization
Word2Vector
Converts words to values for use in NLP tasks, like recommender, named entity recognition, machine translation
Regression
Makes forecasts by estimating the relationship between values
Answers questions like: How much or how many?
Fast Forest Quantile Regression
Predicts a distribution
Poisson Regression
Predicts event counts
Linear Regression
Fast training, linear model
Bayesian Linear Regression
Linear model, small data sets
Decision Forest Regression
Accurate, fast training times
Neural Network Regression
Accurate, long training times
Boosted Decision Tree Regression
Accurate, fast training times, large memory footprint
Recommenders
Predicts what someone will be interested in
Answers the question: What will they be interested in?
Use the Train Wide & Deep Recommender module
Hybrid recommender, both collaborative filtering and content-based approach
SVD Recommender
Collaborative filtering, better performance with lower cost by reducing dimensionality
Clustering
Separates similar data points into intuitive groups
Answers questions like: How is this organized?
K-Means
Unsupervised learning
Anomaly Detection
Identifies and predicts rare or unusual data points
Answers the question: Is this weird?
One Class SVM
Under 100 features, aggressive boundary
PCA-Based Anomaly Detection
Fast training times
Multiclass Classification
Answers complex questions with multiple possible answers
Answers questions like: Is this A or B or C or D?
Multiclass Logistic Regression
Fast training times, linear model
Multiclass Neural Network
Accuracy, long training times
Multiclass Decision Forest
Accuracy, fast training times
One-vs-All Multiclass
Depends on the two-class classifier
One-vs-One Multiclass
Depends on binary classifier, less sensitive to an imbalanced dataset with larger complexity
Multiclass Boosted Decision Tree
Non-parametric, fast training times and scalable
Two-Class Classification
Answers simple two-choice questions, like yes or no, true or false
Answers questions like: Is this A or B?
Two-Class Support Vector Machine
Under 100 features, linear model
Two-Class Averaged Perceptron
Fast training, linear model
Two-Class Decision Forest
Accurate, fast training
Two-Class Logistic Regression
Fast training, linear model
Two-Class Boosted Decision Tree
Accurate, fast training, large memory footprint
Two-Class Neural Network
Accurate, long training times
Image Classification
Classifies images with popular networks
Answers questions like: What does this image represent?
ResNet
Modern deep learning neural network
DenseNet
© 2021 Microsoft Corporation. All rights reserved.
Share this poster: aka.ms/mlcheatsheet

10

10

Regression – Supervised

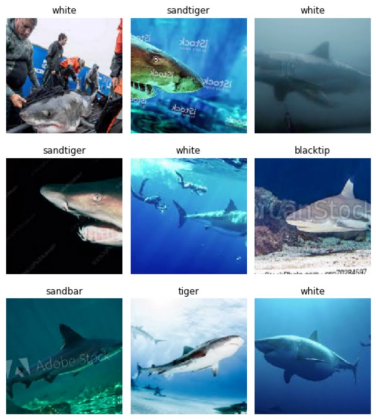
- Also known as Ordinary Least Squares
- Answers the question of how may or how much
- Predicate a continuous number
- Miles per gallon based on vehicle data
- House price based on housing data



11

11

Classification - Supervised



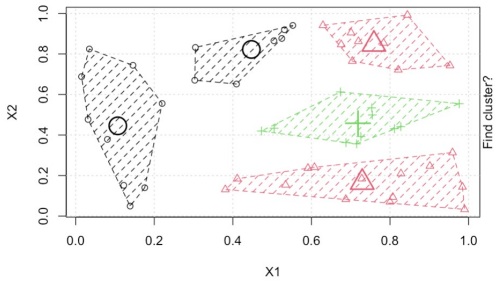
Two class classification Answer a question such as true or false, yes or no, 0 or 1, approved for a loan not approved for a loan

Multiclass Classification Answer a more complex classification with multiple answers or categories

12

12

Clustering - Unsupervised



- Separates similar data into groups
- Answers the question how is the data organized

06- K-means Clustering

13

13

Model Scoring - Regression

- How do you know if the model is doing a good job?
- For Regression there are several numbers to look at, they are all somewhat mathematically related to each other
- R2 (Coefficient of determination in Azure)
  - represents the predictive power of the model as a value between 0 and 1, 0 means random guess
- RMSE (Root Mean Square Error)
  - creates a single value that summarizes the error in the model, in units of the value predicated
- MAE (Mean Absolute Error)
  - measures how close the predictions are to the actual outcomes

14

14

Model Scoring Classification

- Confusion Matrix
- Also know as an error matrix
- TP – True Positive
  - You predicted correctly that it is positive
- TN – True Negative
  - You predicted correctly that it is negative
- FP – False Positive
  - You incorrectly predicted positive
- FN – False Negative
  - You incorrectly predicted it is negative

Predicted Class

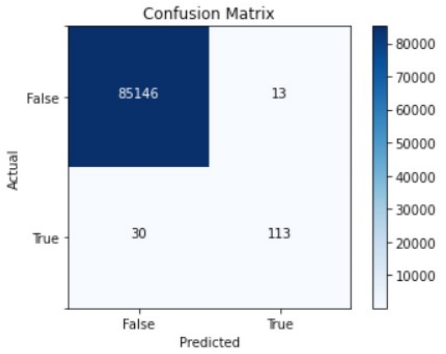
Actual Class	True	False
True	True Positive (TP)	False Negative (FN)
False	False Positive (FP)	True Negative (TN)

15

15

Model Scoring Classification

- Confusion Matrix
- Also know as an error matrix
- TP – True Positive
  - You predicted correctly that it is positive
- TN – True Negative
  - You predicted correctly that it is negative
- FP – False Positive
  - You incorrectly predicted positive
- FN – False Negative
  - You incorrectly predicted it is negative



16

16

Classification - Scoring

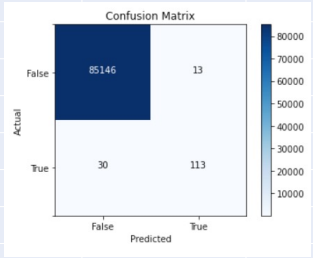
- **Accuracy** - (TP+TN)/(TP+TN+FP+FN)
  - the goodness of a classification model as the proportion of true results to total cases
- **Precision** - TP/(TP+FP)
  - proportion of true results over all positive results
- **Recall** - TP/(TP+FN)
  - fraction of the total amount of relevant instances that were actually retrieved
- **F1-Score** – 2*(Precision*Recall)/(Precision+Recall)
  - weighted average of precision and recall between 0 and 1, where the ideal F1 score value is 1.
- **AUC** – (Area Under the Curve)
  - Measures the quality of the model's predictions irrespective of what classification threshold is chosen. This metric is useful because it provides a single number that lets you compare models of different types

17

17

Classification Metric in Practice

Predict/Actual	1	0		
1	113	30		
0	13	85,146		
			ACCURACY	1.00
			PRECISION	0.90
			RECALL	0.79
			F1SCORE	0.84



18

Anatomy of an ML Training Process

1. Retrieve the data
2. Visualize the data look for outliers, bad data, missing data
3. Normalize the data, drop columns that have nothing to do with the target variable, or low to zero correlation
4. One-Hot encode categorical variables
5. Split training and test dataset, don't forget to keep a hold out(validation) dataset too
6. Train the model on the training dataset and adjust hyperparameters
7. Score the model on the test dataset
8. Repeat

1. Retrieve the data

- Pandas is the package typically used for reading data into a dataframe
  - Clipboard
  - Csv
  - Excel
  - Feather
  - Html
  - Json
  - Orc

```
In [14]: #Read the data in from somewhere
         epa = pd.read_csv('https://raw.githubusercontent.com/sqlshep/SQLShepBlog/master/data/epaMpg.csv')
```

Take a look at the data

- head(10) display the first ten rows, equally you could use tail(10)

```
In [15]: epa.head(10)
```

Out[15]:

	RowNumber	Represented.Test.Veh.Make	Model	Vehicle.Type	HorsePower	Cylinders	Tested.Transmission.Type.Code	Tested.Transmission.Type	Gears	D
0	1	Aston Martin	Rapide S	Car	552	12	SA	Semi-Automatic	8	
1	2	Aston Martin	Vanquish	Car	568	12	SA	Semi-Automatic	8	
2	3	BENTLEY	Continental GT	Car	616	12	SA	Semi-Automatic	8	
3	4	BENTLEY	Continental GT	Car	616	12	SA	Semi-Automatic	8	
4	5	BMW	230i Convertible	Car	248	4	SA	Semi-Automatic	8	
5	6	BMW	230i Coupe	Car	248	4	M	Manual	6	
6	7	BMW	230i Coupe	Car	248	4	SA	Semi-Automatic	8	
7	8	BMW	230i xDrive Convertible	Car	248	4	SA	Semi-Automatic	8	
8	9	BMW	230i xDrive Coupe	Car	248	4	SA	Semi-Automatic	8	
9	10	BMW	320i	Both	181	4	A	Automatic	8	

21

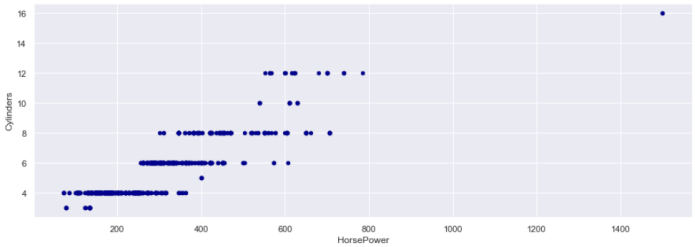
21

2. Visualize the data

- Pandas comes with basic visualization features

```
In [32]: epa.plot.scatter(x='HorsePower',
                          y='Cylinders',
                          c='DarkBlue',
                        figsize=(15,5))
```

Out[32]: <AxesSubplot: xlabel='HorsePower', ylabel='Cylinders'>



22

22

3/4. Normalize the data, one hot-encode

- Remove unneeded columns, fix column headers, one-hot encode categorical data

```
epa['Tested_Transmission_Type_Code']= epa['Tested_Transmission_Type_Code'].astype('category')

epa['Drive_System_Code']= epa['Drive_System_Code'].astype('category')
```

```
#One hot encode categories
epa = pd.get_dummies(epa)
```

```
print(epa.shape)
epa
```

```
(1034, 20)
```

Tested_Transmission_Type_Code_A	Tested_Transmission_Type_Code_AM	Tested_Transmission_Type_Code_AMS	Tested_Transmission_Type_Code_CVT	Tested_Transm
0	0	0	0	
0	0	0	0	
0	0	0	0	
0	0	0	0	
0	0	0	0	
...	...	...	...	
0	0	0	0	
0	0	0	0	
0	0	0	0	
0	0	0	0	
0	0	0	0	

23

23

5. Split training and test datasets

- Sklearn Train_test_split(array of matrices, test_size)
  - Inputs are numpy arrays, dataframes, sparse matricies
  - Test_size is the percentage of rows to populate test dataframe with(X_test, y_test)

```
In [52]: # Create the training dataset for scikit learn, you will need all
         # varialbes except the label you are trying to predict
         epa_X = epa.iloc[:, epa.columns !='FuelEcon']
```

```
In [53]: # You will also need a dataset the the target varialbe
         epa_y = epa.iloc[:, epa.columns =='FuelEcon']
```

```
In [40]: # Split the training and test set
         X_train, X_test, y_train, y_test = train_test_split(epa_X, epa_y, test_size=0.20)
```

```
In [41]: print(X_train.shape, X_test.shape, y_train.shape, y_test.shape )
         (827, 1) (207, 1) (827, 1) (207, 1)
```

24

24

6. Train the model

- Import the model from ml package
- Create the model object
- Run fit, this creates the estimator

```
In [59]: from sklearn import linear_model

In [61]: epa_lm = linear_model.LinearRegression()

In [62]: epa_lm.fit(X_train, y_train)
Out[62]: LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=False)
```

25

25

7. Score the model

```
In [75]: import math
         model_metrics = pd.DataFrame(columns=["Model", "MSE", "RMSE", "R2"])

         def metrics(model, y, y_hat):

             model_metrics.loc[-1] = {"Model" : model,
                       "MSE" : mean_squared_error(y, y_hat),
                       "RMSE" : math.sqrt(mean_squared_error(y, y_hat)),
                       "R2" : r2_score(y, y_hat)}

             model_metrics.index = model_metrics.index + 1
             return model_metrics

         #metrics("PCA Forest",y_test, epa_pca_y_pred)

In [63]: epa_y_pred = epa_lm.predict(X_test)

In [76]: metrics("linear_model",y_test, epa_y_pred)
```

Out[76]:

	Model	MSE	RMSE	R2
0	linear_model	34.504881	5.874086	0.612347

- To score the model, you use the freshly created estimator and make predictions on the test dataset you created in prior step
- Save the results to a list
- Compare the saved results to the original ground truth

26

26

8. Repeat

- Now that you have a model created you will iterate through the process tuning hyperparameters
- Hyperparameters are not directly learnt within the estimator, they are arguments passed through the constructor
- There are tools built into SKLearn to help.
- Grid search – exhaustive search of passed parameters, (slow)
- Random Search – a randomized search over parameters is passed in
- Bayesian Search – Uses Stochastic Gradient Descent to optimize the next parameter based on the error from the last model creation

27

27

Thank you for attending the MLADS Conference and helping to build a strong community

To find recordings, presentations, and other resources from the event, go to: http://aka.ms/november2022mlads

More information about the AI & ML Connected Community: http://aka.ms/aiml-cc

We want your feedback!

We read each evaluation and incorporate your comments into future MLADS events.
Both sessions and event evaluations will be available within the MyMLADS event portal during the conference dates—to access them, navigate to the **Evaluations tab**.

We request that session evaluations are submitted following each day of programming.

28

28


Q&A
This slide is required—do not delete—please read the notes for this slide, then delete this text box
29

29


Microsoft
© Copyright Microsoft Corporation. All rights reserved.

30