SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

Established in collaboration with MIT

Feature Selection and Time Series

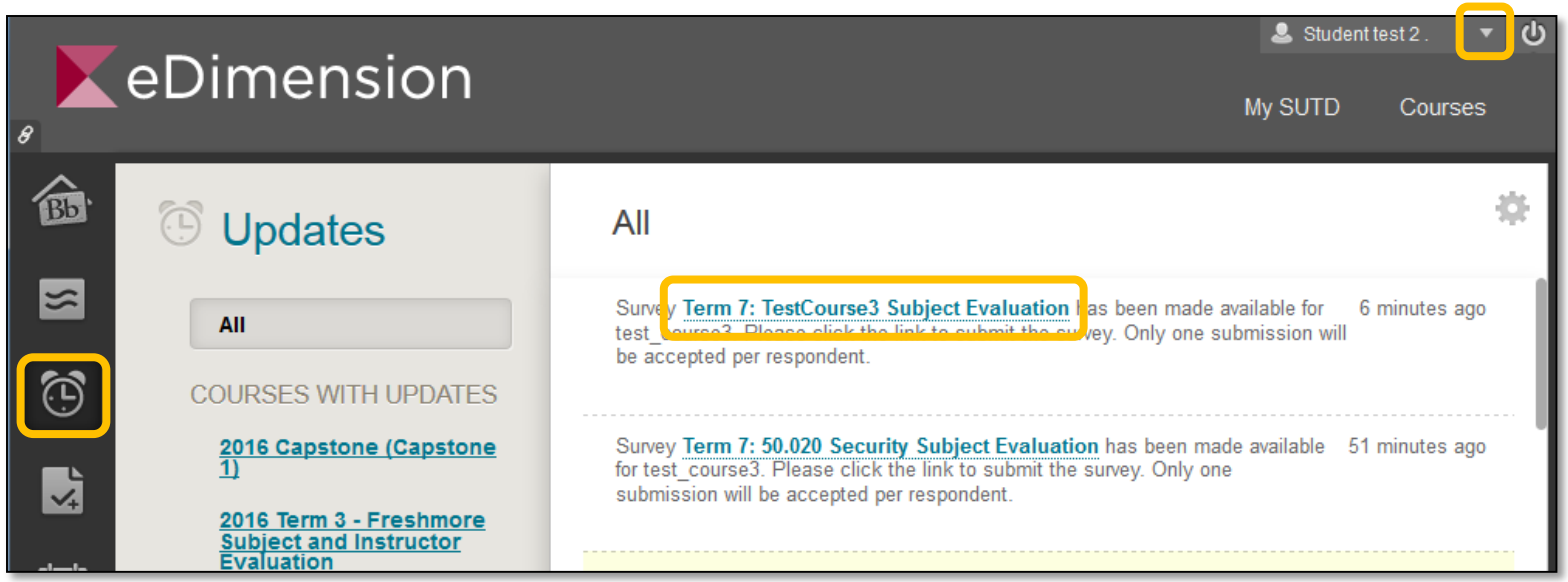
PROF. K. H. LIM

50.038 Computational Data Science

Reminder: Complete Survey

- Complete your ***Mid Term Subject & Instructor Survey*** on eDimension

  1. Click on the *Global Navigation Menu arrow* > *Updates (clock) icon*. The survey links are listed under All section.
  2. Click on a survey link to participate e.g. *Term 7: TestCourse3.*.



Python Scikit-learn

- Machine learning library based on Python
- Contains functionalities for data pre-processing, classification, clustering, etc

Twitter Dataset

- We will use a smaller Twitter dataset (for faster computation), based on a 10% sample of the 1.6M tweets in Lab 3
  - Download from eDimensions
  - Alternatively, create your own sample with data.sample(n=160000)

- This dataset comprises 160k tweets with various columns, we will make use of the first and last column (sentiment label and tweet text)
  - For sentiment, a value of 4 = positive and 0 = negative

Load Packages

- Import relevant packages

```
1  from sklearn.feature_extraction.text import CountVectorizer
2  from sklearn.naive_bayes import MultinomialNB
3  from sklearn.pipeline import Pipeline
4  from sklearn.model_selection import train_test_split
5  from sklearn.preprocessing import FunctionTransformer
6  from sklearn import metrics
7  from sklearn.metrics import accuracy_score
8  from sklearn.feature_selection import SelectKBest, SelectPercentile
9  from sklearn.feature_selection import chi2, f_classif, mutual_info_classif
10 import pandas as pd
11 import numpy as np
```

Load Dataset and Check

In [2]:
```
# load in training/test set
data = pd.read_csv('tweets.160k.random.csv', encoding='utf-8')
data.head()
```

Out[2]:

	label	id	date	query	user	text
0	4	1985770747	Sun May 31 17:44:25 PDT 2009	NO_QUERY	vozabala	Getting ready for another week of fun and game...
1	0	2322735567	Wed Jun 24 23:10:08 PDT 2009	NO_QUERY	liannecab	http://twitpic.com/8cp6u - I want it, sooo bad
2	0	1972997427	Sat May 30 10:16:49 PDT 2009	NO_QUERY	nadhirarchangel	iloveyousincethe1stgradeitsthefirsttimewemet ...
3	0	2230992481	Thu Jun 18 17:53:46 PDT 2009	NO_QUERY	doughamlin	@extendr I can add :skype links but :aim links...
4	4	2053227537	Sat Jun 06 03:46:32 PDT 2009	NO_QUERY	Mariallama	just woke up at to rain. . . on the plus side ...

In [3]:
```
data['label'].value_counts()
```

Out[3]:
```
4    80259
0    79741
Name: label, dtype: int64
```

Define Pipeline Components

- Train/test set, column extractor, features, classifier

```
# build a pipeline components for uni-grams and bi-grams, using a 80:20 train/test split
labels = data['label']
x_train, x_test, y_train, y_test = train_test_split(data, labels, test_size=0.2)

getTweetCol = FunctionTransformer(lambda x: x['text'], validate=False) # extract tweets
tfVect = CountVectorizer(stop_words='english', lowercase=True, ngram_range=(1,2))
mnbClf = MultinomialNB()
```

Train, Test and Evaluate

- Build our pipeline for training, testing and evaluating
  - Similar to lab 3
- For this lab, we focus on the accuracy metric

```
clf_tf = Pipeline([('getTweets', getTweetCol), ('vect', tfVect), ('clf', mnbClf)])
clf_tf.fit(x_train, y_train)
predicted = clf_tf.predict(x_test)
print(accuracy_score(y_test, predicted))
```

Feature Selection

- Adding feature selection to our pipeline

```
1  feaSelect = SelectPercentile(chi2, percentile=5)
2
3  clf_tf = Pipeline([('getTweets', getTweetCol), ('vect', tfVect),
4                     ('feaSelect', feaSelect), ('clf', mnbClf)])
5  clf_tf.fit(x_train, y_train)
6  predicted = clf_tf.predict(x_test)
7
8  print(accuracy_score(y_test, predicted))
```

Feature Selection

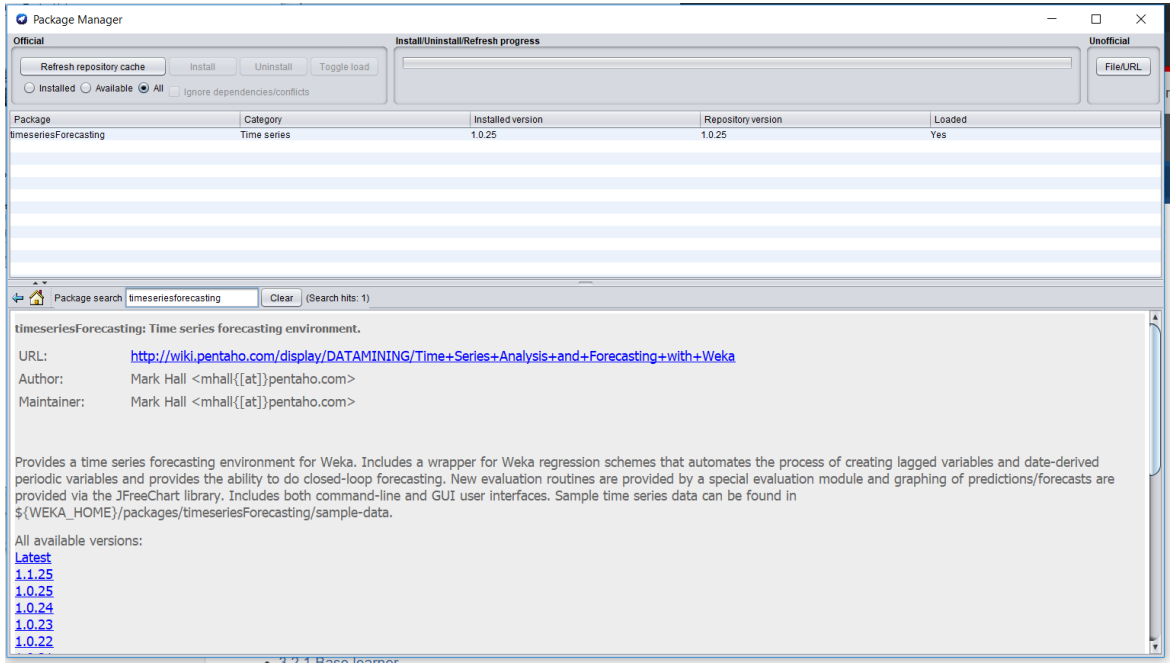
- Feature selection by top-k features or percentile
  - See documentation for sklearn.feature_selection.SelectKBest and sklearn.feature_selection.SelectPercentile
- Various scoring functions (ANOVA F-value, Mutual information, Chi-square)
  - See documentation for f_classif, chi2, mutual_info_classif

Exercise

1. Similar to Lab 3 Exercise 6, build a Naïve Bayes classifier for this sentiment task using uni-grams and bi-grams
2. Extend our classifier to include different types of feature selection
   - Try feature selection by ***SelectKBest*** and ***SelectPercentile***
   - Try the ***f_classif*** and ***chi2*** scoring functions

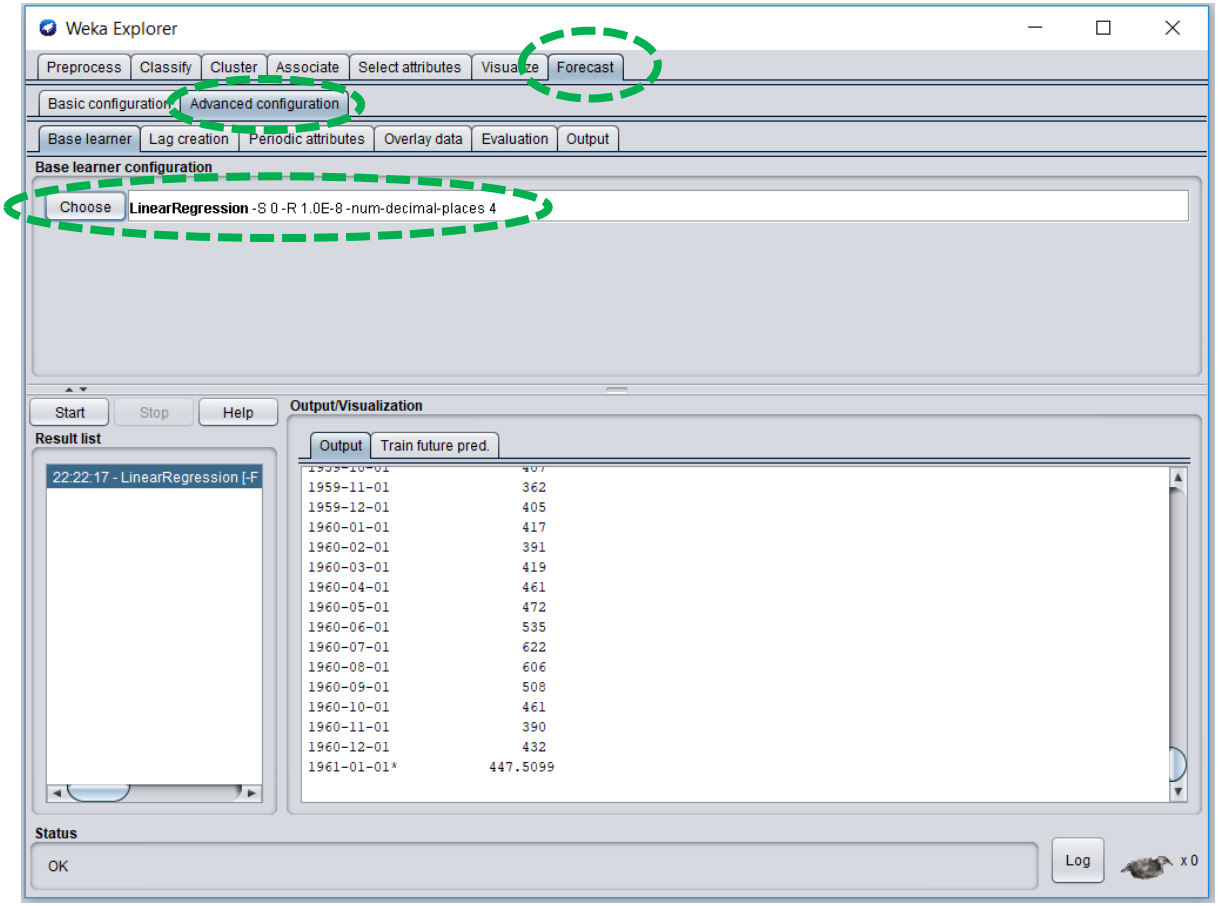
Weka timeseriesForecasting

- Install Weka package “timeseriesForecasting”
  - Go to Tools → Package Manager → Search for “timeseriesForecasting”



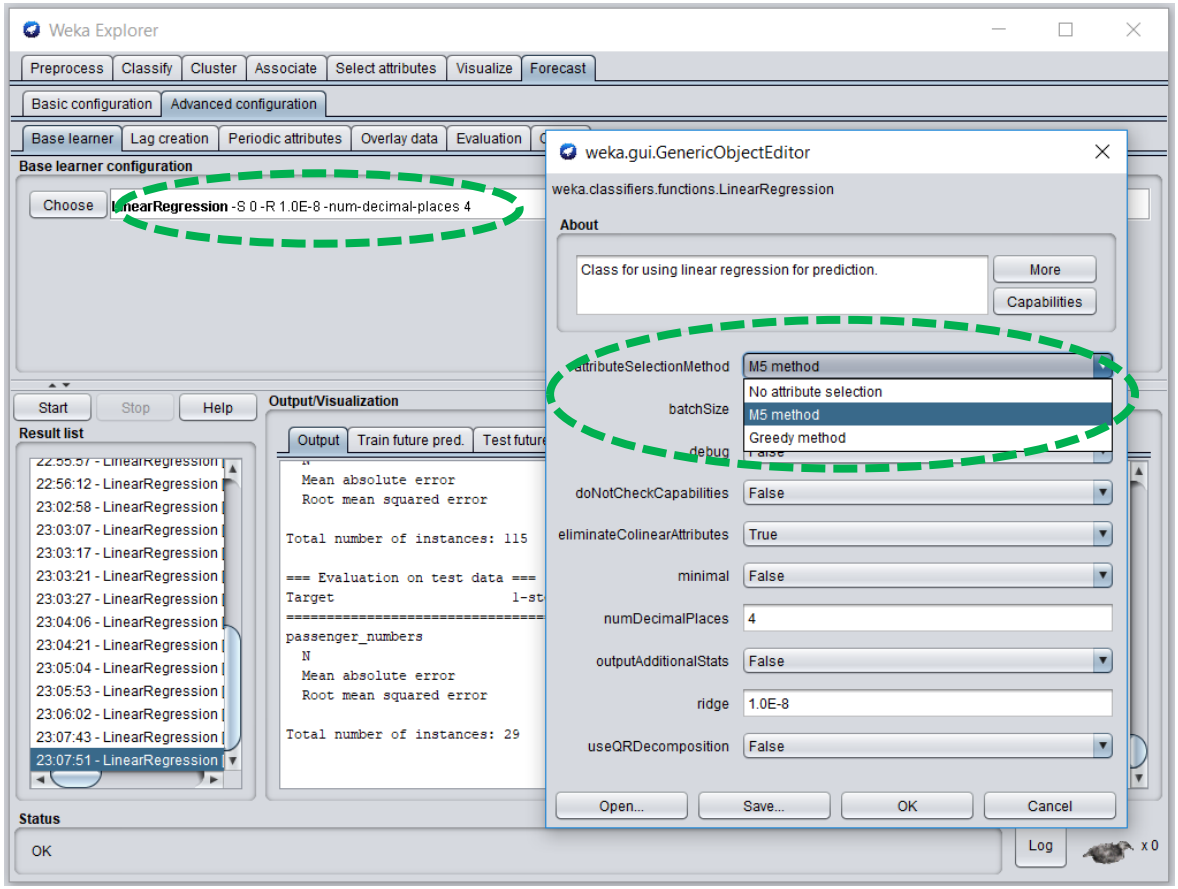
Predicting Time Series

- Option to select from various algorithms
- Forecast→Advanced configaration→Base learner→Choose



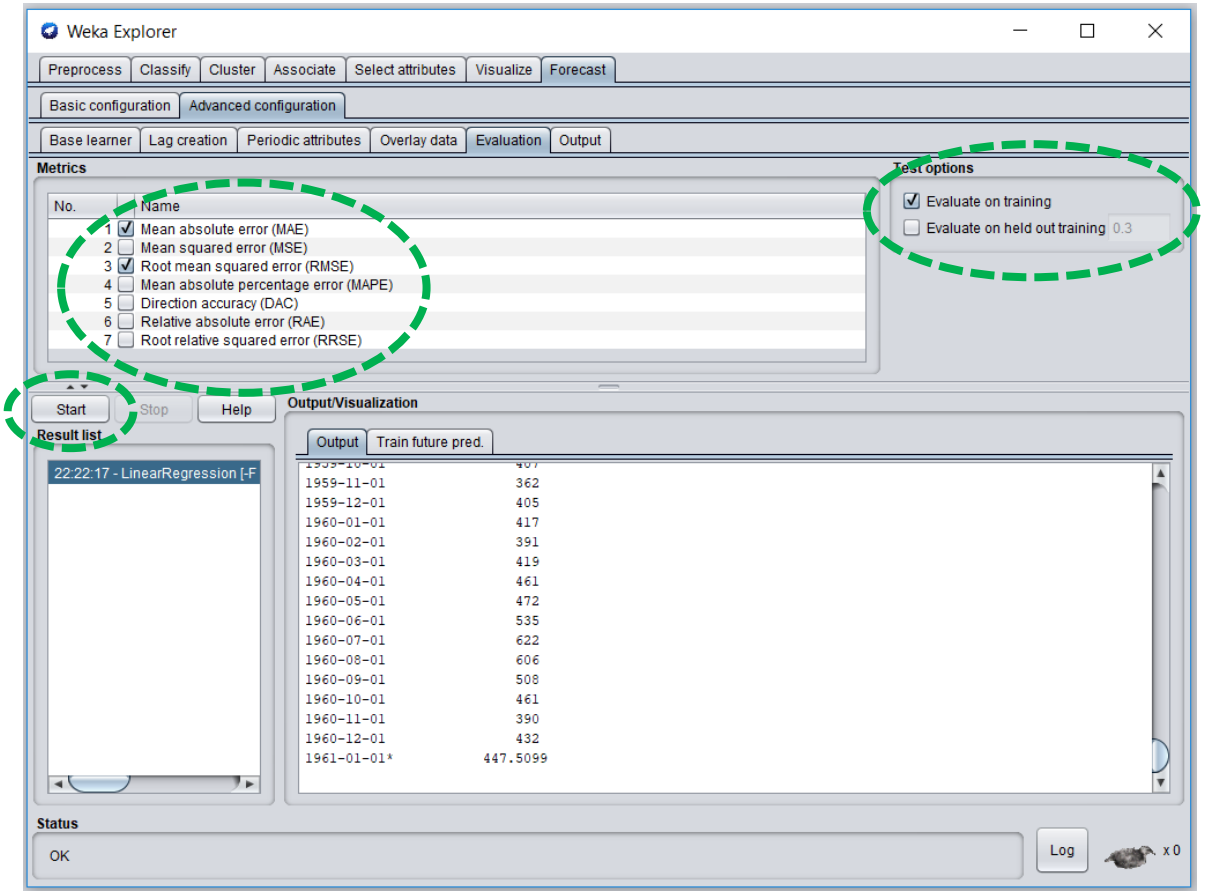
Feature Selection

- Option to choose feature selection method
- Choose→LinearRegression→attributeSelectionMethod



Evaluation

- Select different ways of evaluation
- Various evaluation metrics available



Exercise

3. Load in the ./weka-3.8/data/airline.arff dataset
4. Using LinearRegression, evaluate on the training set and observe the results in terms of MAE and RMSE
5. Repeat Step 4, but evaluate on 20% of the dataset. How does the MAE and RMSE compare to those from Step 4? Why is it so?
6. Repeat Step 5, but evaluate LinearRegression without attribute (feature) selection. What is the performance now?