Analysis signals from sleep with Data mining technique for predict sleep stages

การวิเคราะห์คลื่นสัญญาณจากการนอน ด้วยเทคนิคการเรียนรู้ของเครื่องเพื่อทำนายระดับการนอนหลับ

Narongrit Sridonthong 5710210119
Wanchai Nuptnit 5710210389

Advisor
Asst. Prof. Thakerng Wongsirichot

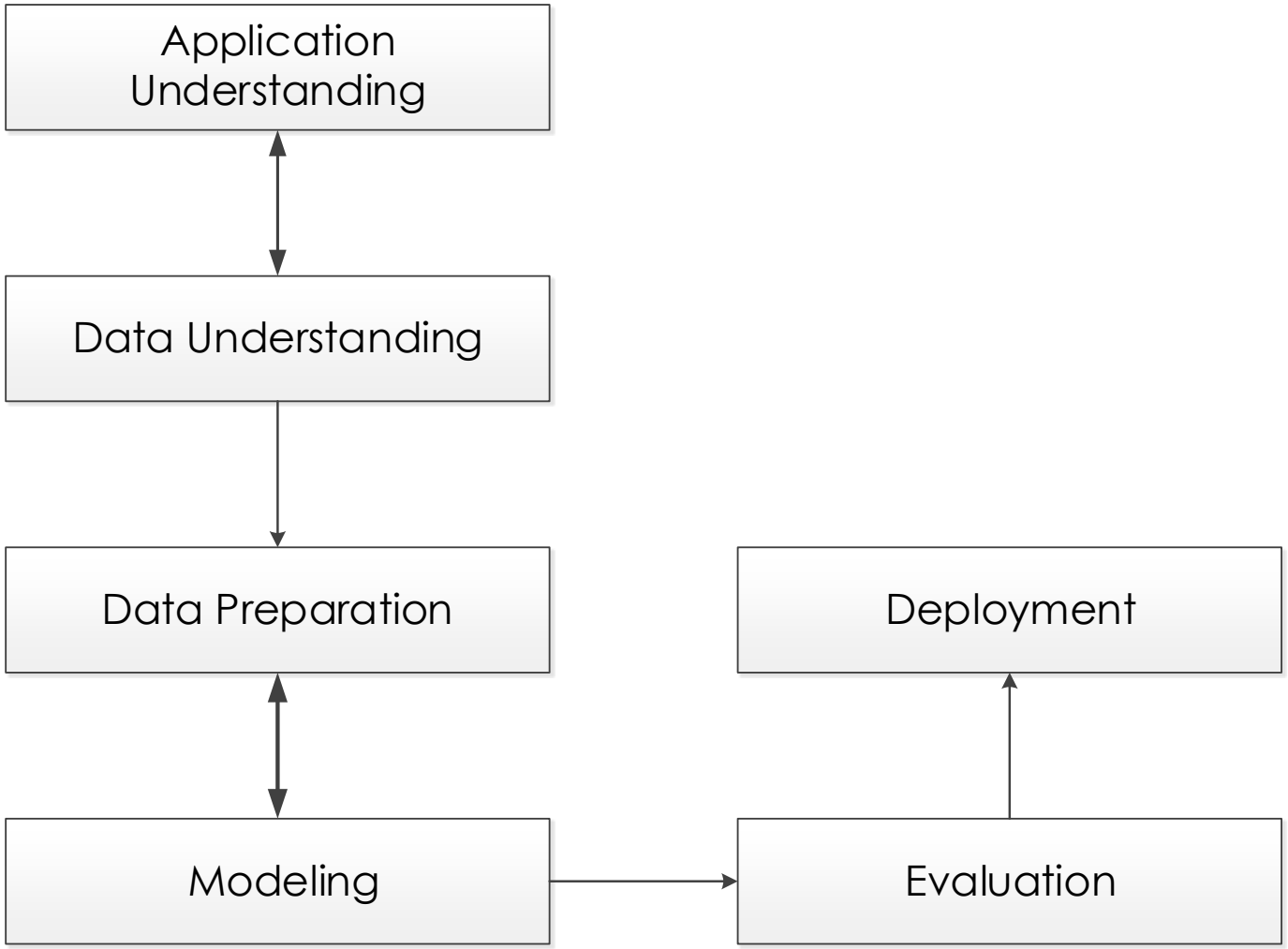
Scope

- Find the best model that can classify sleep stages from data signals.
- Analysis and evaluate result of model
- Improve model for best classification

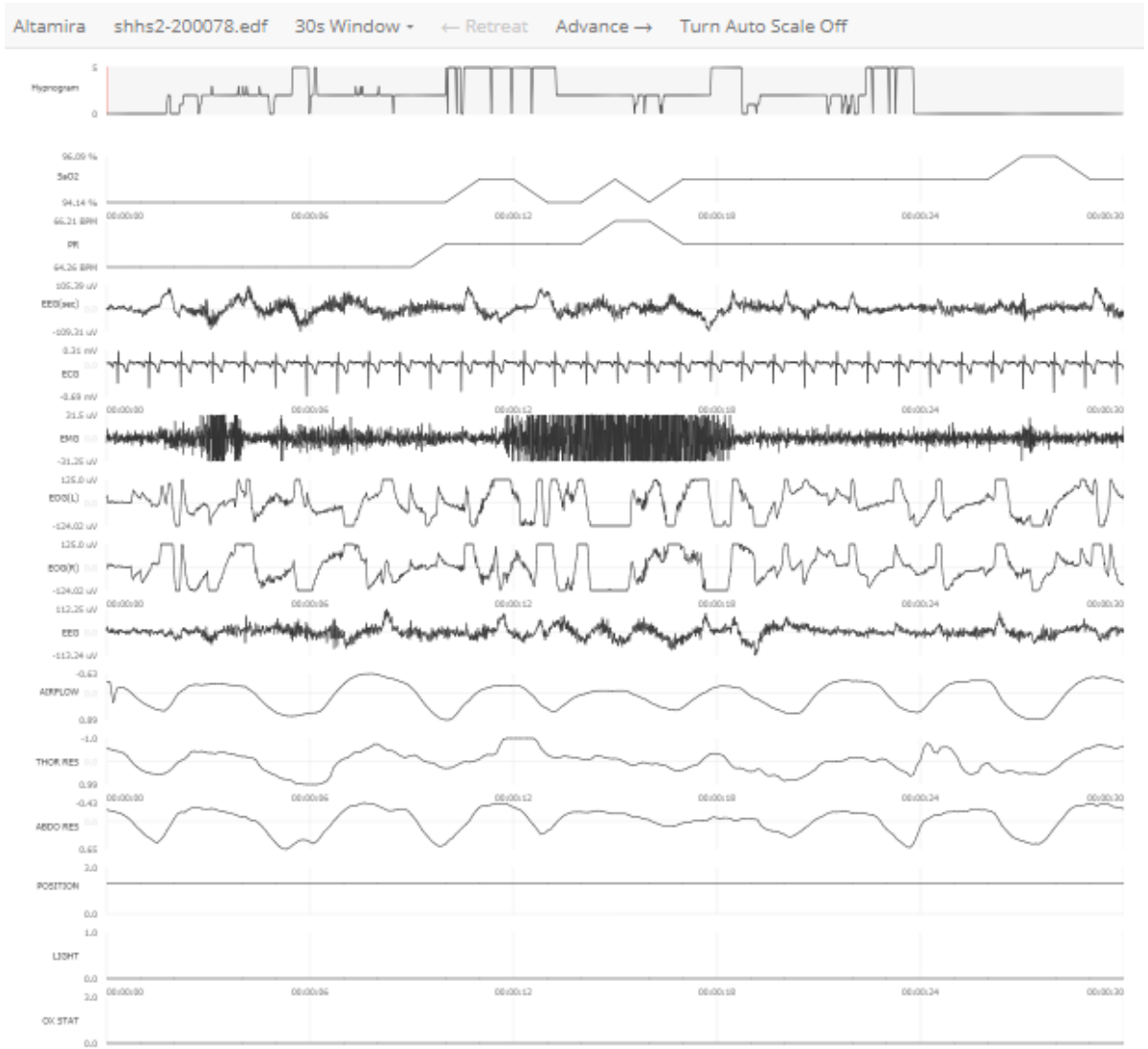
Objective

- To develop methodology for sleep stages classification with physical signals
- To apply the model for automatic analyze sleep stages and can easy to analyze with sleep specialists.

CRISP-DM


Application Understanding
Data Understanding
Data Preparation
Modeling
Evaluation
Deployment

Data Understaning



Channel	Sampling Rate (Hz)
SaO2	1
PR	1
EEG (sec)	128
ECG	256
EMG	128
EOG(L)	64
EOG(R)	64
EEG	128
Airflow	8
Thor RES	8
Abdo RES	8
Position	1
Light	1
OX STAT	1

128 values per second

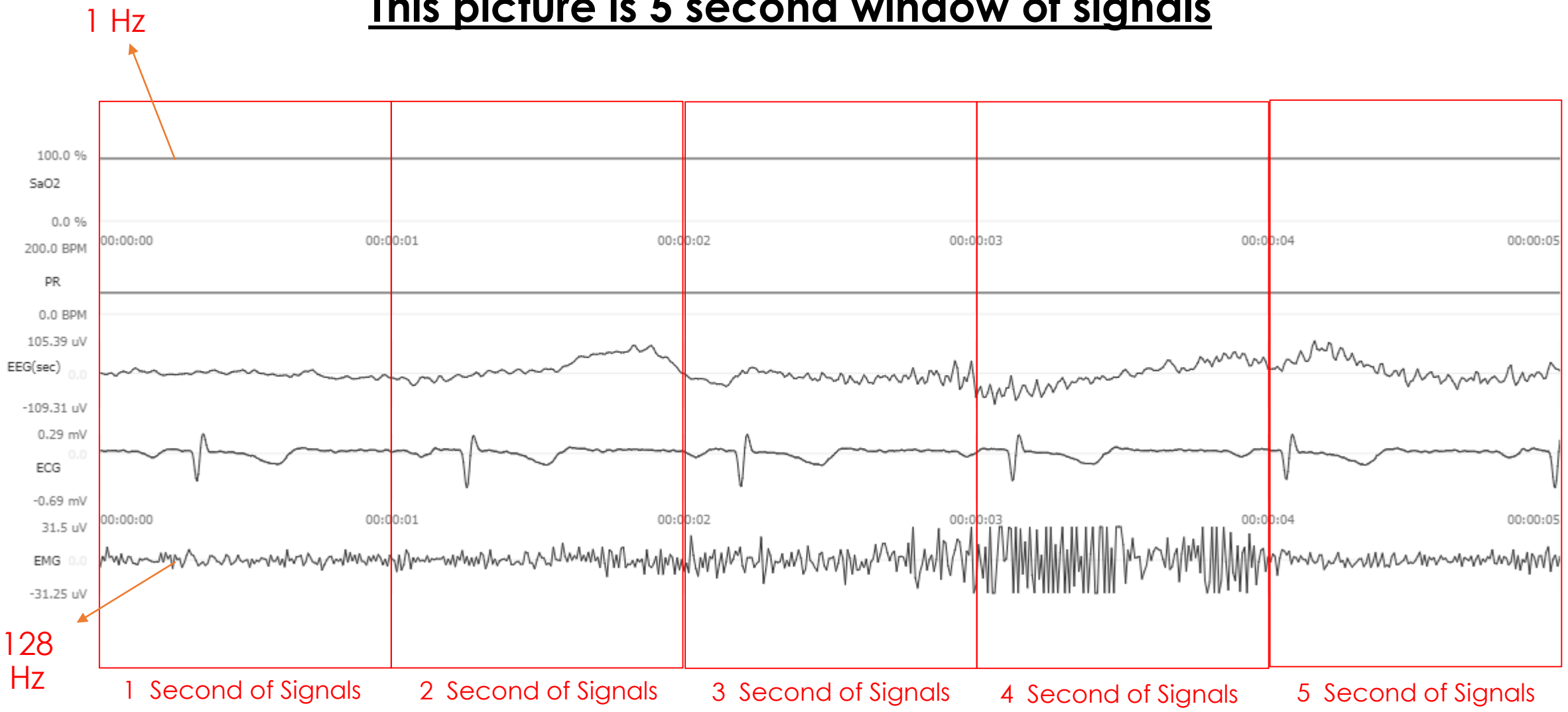
64 values per second

8 values per second

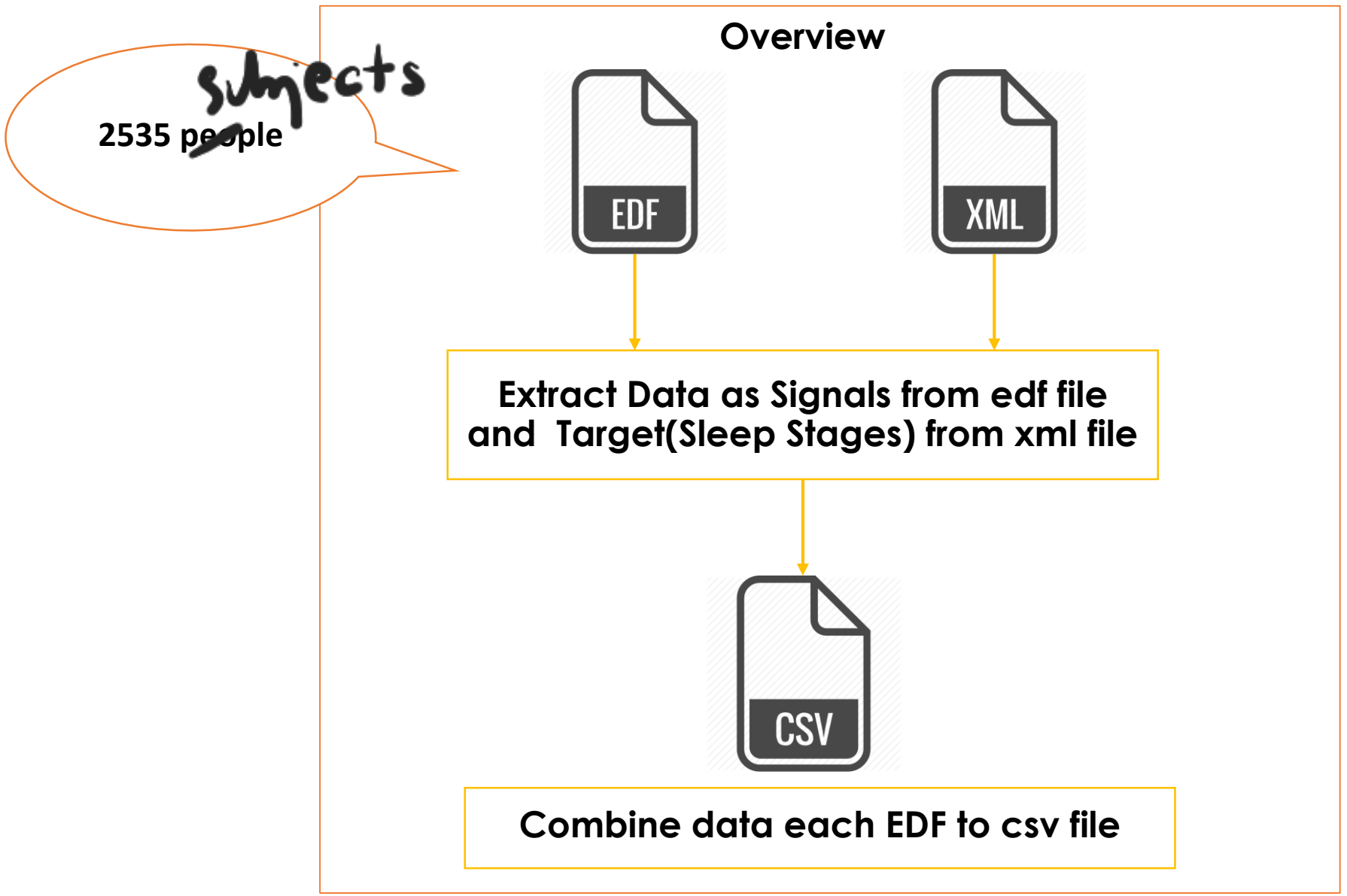
- f = 1.0 Hz
  T = 1.0 s
- f = 2.0 Hz
  T = 0.5 s

Yo dataset? + Reference

This picture is 5 second window of signals

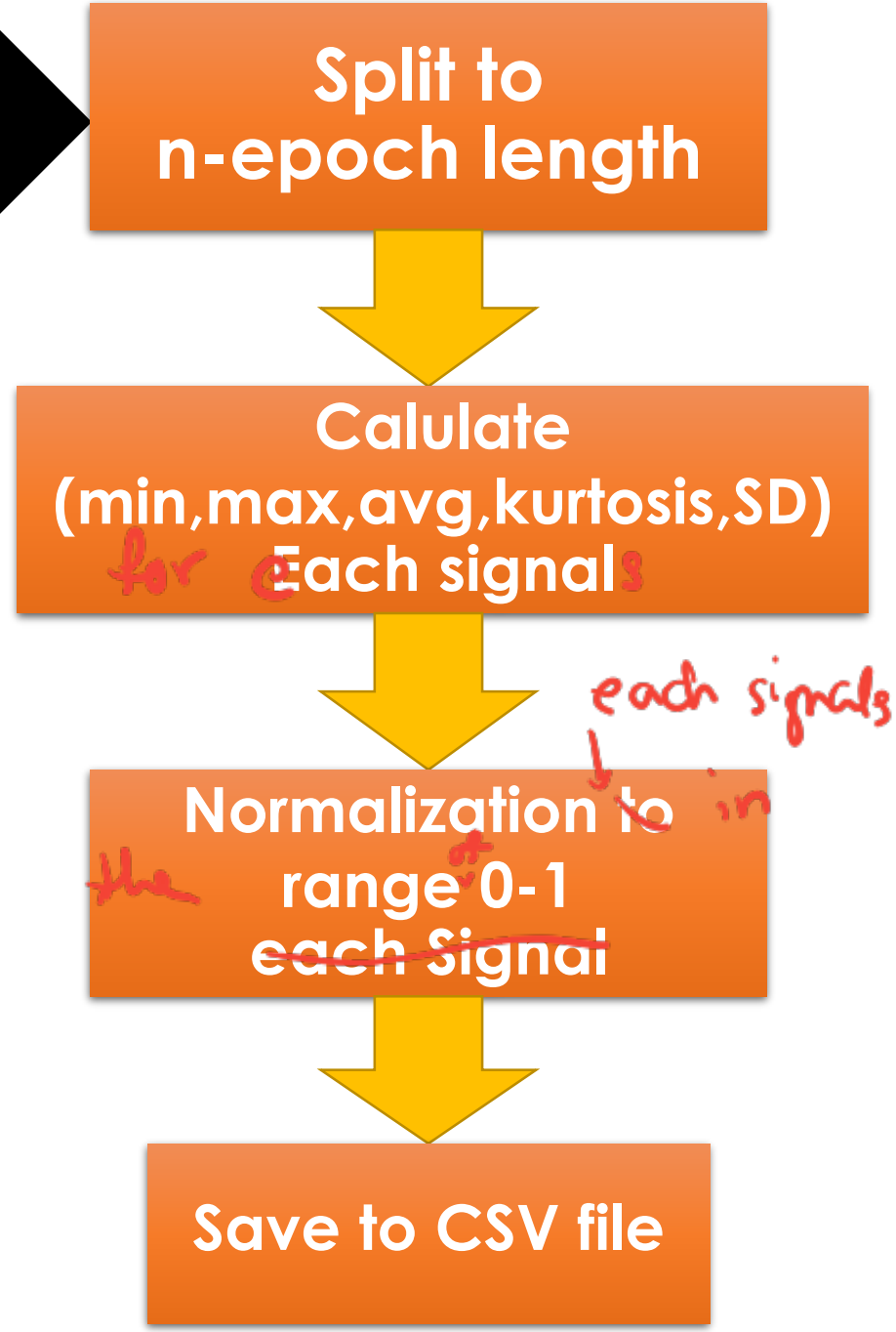


Data Preprocessing



```
Select C:\WINDOWS\system32\cmd.exe
C:\Users\chai_\Google Drive\1_2560 (1)\308- Project2\edf\!!##Preprocess>python ReadEDF.py
Enter path:C:\Users\chai_\Google Drive\1_2560 (1)\308- Project2\edf\shhs1
0 shhs2-200077.edf
1 shhs2-200078.edf
Select file number:0
Sample Frequencies: [  1   1 128 256 128  32  32 128   8   8   8   1   1   1]
Length Samples
 [   46290    46290  5925120 11850240  5925120  1481280  1481280  5925120
   370320   370320   370320    46290    46290    46290]
(Sa02) [ 95.11558709  96.09216449  96.09216449  96.09216449  96.09216449]

(PR) [ 76.37140459  77.34798199  79.3011368   81.2542916   83.2074464 ]

(EEG(sec)) [ -2.45098039  -8.33333333 -16.17647059 -26.96078431 -16.17647059]

(ECG) [ 0.21078431  0.20098039  0.19117647  0.19117647  0.18137255]

(EMG) [-2.59411765 -1.85294118 -3.08823529  5.31176471 -8.77058824]

(EOG(L)) [ 11.2745098   21.07843137 -14.21568627  -8.33333333 -10.29411765]

(EOG(R)) [-23.03921569 -16.17647059   5.39215686   8.33333333   6.37254902]

(EEG) [ -6.37254902  16.17647059  19.11764706  10.29411765   5.39215686]

(AIRFLOW) [-0.90588235 -0.49019608 -0.09019608  0.23921569  0.4745098 ]

(THOR RES) [-0.28627451 -0.24705882 -0.16078431 -0.09019608  0.02745098]

(ABDO RES) [-0.60784314 -0.52156863 -0.46666667 -0.29411765  0.27843137]

(POSITION) [ 3.  3.  3.  3.  3.]
```

Sample Data from EDF file



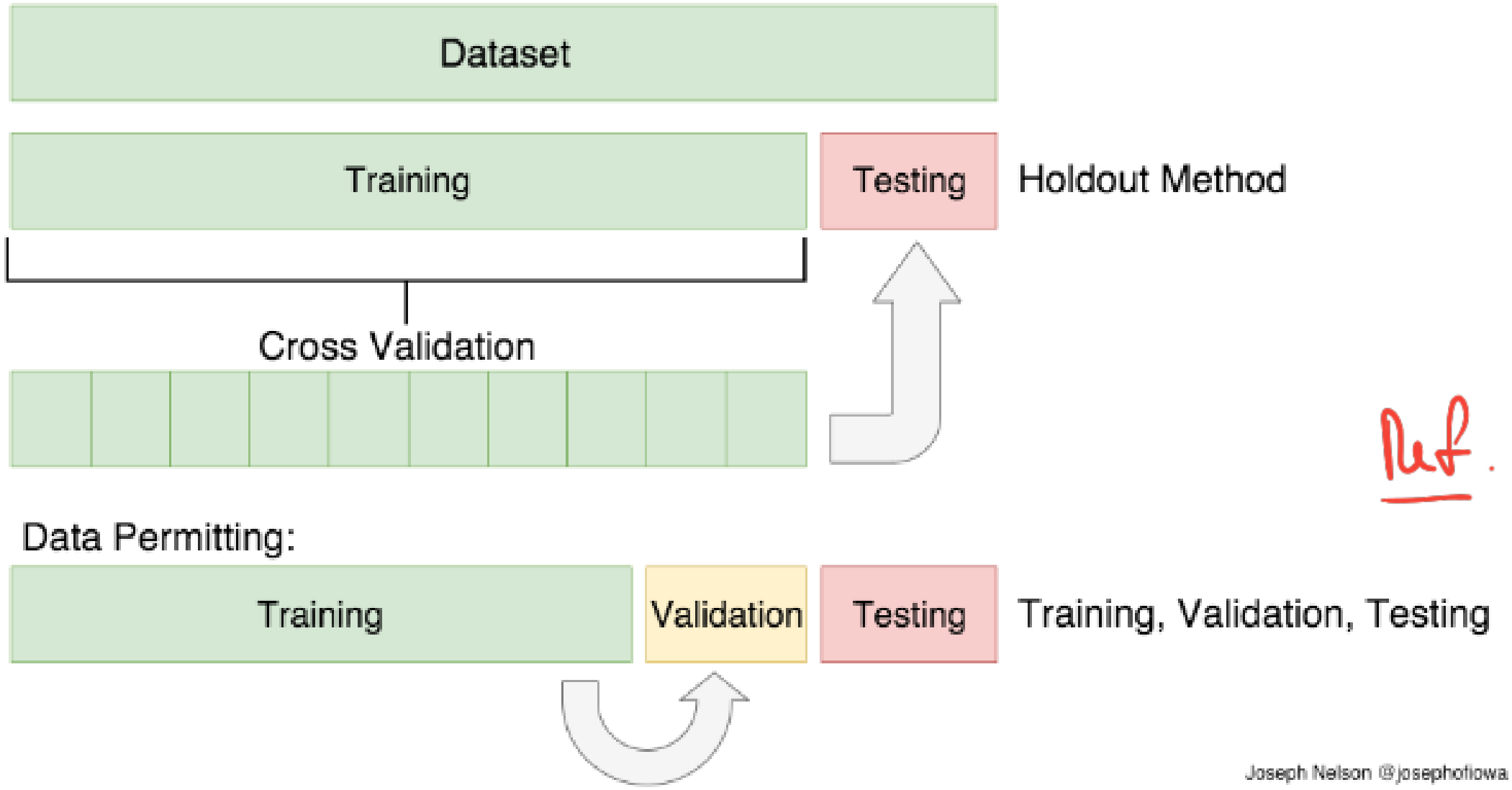
Example of trainset (CSV file)

training set

1	subject	epoch	SaO2_min	SaO2_max	SaO2_avg	SaO2_SD	SaO2_kurt	PR_min	PR_max	PR_avg	PR_SD	PR_kurtosi	position_m	position_m	position_av	position_SI	position_ku	light_m
2	shhs2-205399	501	0.988933	1	0.992328	0.010086	0.091962	0.694035	0.355817	0.671713	0.028071	0.0512	0.333333	0.333333	0.333333	0	0	
3	shhs2-201859	1121	0	0	0	0	2.78E-17	0	0	0	5.61E-19	0.03567	1	1	1	0	0	
4	shhs2-200116	965	0.97923	0.989722	0.986555	0.015958	0.038453	0.724696	0.452776	0.769169	0.060882	0.048514	1	1	1	0	0	
5	shhs2-201610	947	0.928282	0.939839	0.940269	0.0215	0.017494	0.666656	0.621256	0.657517	0.02262	0.037499	0.666667	0.666667	0.666667	0	0	
6	shhs2-200680	909	0	0	0	0	2.78E-17	0	0	0	0	0	1	1	1	0	0	
7	shhs2-200796	1052	0	0	0	0	2.78E-17	0	0	0	6.95E-19	0.03567	0	0.333333	0.016667	0.171935	1	
8	shhs2-204680	564	0.990039	0.782609	0.977154	0	2.78E-17	0.627589	0.095238	0.504101	0.016331	0.091158	0	0	0	0	0	
9	shhs2-203167	1050	0.959161	0.969426	0.972119	0.010903	0.045088	0.594407	0.305962	0.586498	0.021944	0.10001	1	1	1	0	0	
10	shhs2-204086	685	0.947581	0.937622	0.945414	0.010398	0.000661	0.711731	0.725639	0.72495	0.040782	0.085614	0	0	0	0	0	
11	shhs2-203807	768	0.979082	0.990137	0.990782	0.018423	0.022347	0.726571	0.285433	0.581414	0.006909	0.053506	0	0	0	0	0	
12	shhs2-205398	899	0.95875	0.931555	0.950462	3.05E-16	2.78E-17	0.730637	0.567043	0.6936	0.013414	0.075394	0.666667	0.666667	0.666667	0	0	
13	shhs2-203966	1047	0	0	0	0	2.78E-17	0	0	0	0	0	1	1	1	0	0	
14	shhs2-200813	754	0.948689	0.979082	0.975266	0.022109	0.148728	0.691555	0.640039	0.755525	0.177968	0.0264	0.666667	0.666667	0.666667	0	0	
15	shhs2-203495	1000	0.939239	0.980275	0.953333	0.040108	0.046658	0.640198	0.751398	0.70299	0.083808	0.05041	0.333333	0.333333	0.333333	0	0	
16	shhs2-201919	270	0.959306	0.979082	0.962043	0.031645	0.023067	0.577854	0.417813	0.443588	0.01693	0.052623	0.666667	0.666667	0.666667	0	0	
17	shhs2-200994	423	0.969122	0.969122	0.969122	6.45E-16	2.78E-17	0.64424	0.320625	0.425838	0.01808	0.15716	1	1	1	0	0	
18	shhs2-204312	5	0	0.971635	0.805962	0.719504	0.118154	0	0.350929	0.382092	0.214153	0.136056	0.333333	1	0.733333	0.413503	0.103031	
19	shhs2-204904	949	0.990137	0.619806	0.974324	0	2.78E-17	0.784215	0.225154	0.593733	0.021706	0.065232	0	0	0	0	0	
20	shhs2-205320	229	0.959564	0.960882	0.965821	0.006928	0.172145	0.456135	0.368844	0.408382	0.004047	0.172145	0.666667	0.666667	0.666667	0	0	
21	shhs2-200226	1120	0	0	0	0	0	0	0	0	0	0	0.666667	0.666667	0.666667	0	0	
22	shhs2-203080	787	0.928282	0.931555	0.93411	0.0162	0.031608	0.733323	0.310849	0.515798	0.015603	0.061839	0.333333	0.333333	0.333333	0	0	
23	shhs2-205026	655	0.969122	0.969426	0.971834	0.011117	0.000661	0.466783	0.325513	0.455174	0.010726	0.02666	0.666667	0.5	0.5	0	0	
24	shhs2-200740	1302	0.979444	0.672131	0.924443	0.013713	0.011327	0.584408	0.123307	0.268712	0.039229	0.056928	0.5	0.333333	0.344828	0	0	

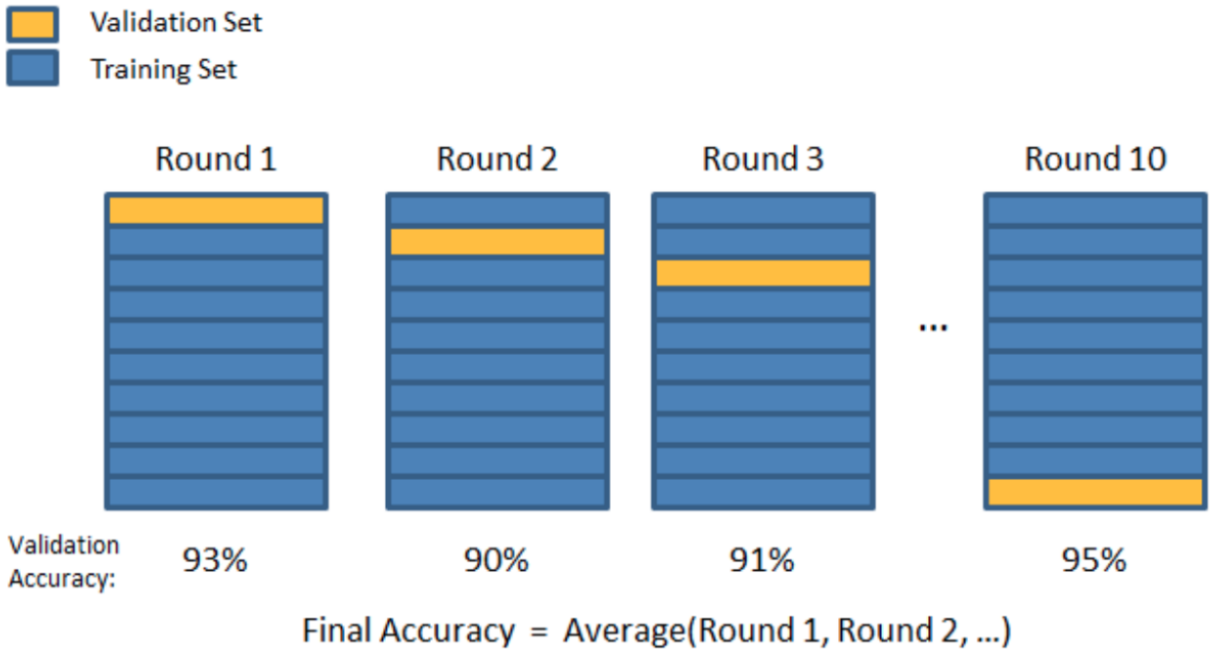
Modeling

- **Supervised learning**
  - It is the machine learning task of inferring a function from *labeled training data*.
- **Classification Model**
  - Naïve Bayes
  - Nearest Neighbor
  - Neuron Network
  - Random Forest
  - Decision Tree
  - Support vector machines



Visual Representation of Train/Test Split and Cross Validation. H/t to my DSI instructor, Joseph Nelson!

K-Folds Cross Validation



- In K-Folds Cross Validation we split our data into k different subsets (or folds). We use k-1 subsets to train our data and leave the last subset (or the last fold) as test data. We then average the model against each of the folds and then finalize our model. After that we test it against the test set.

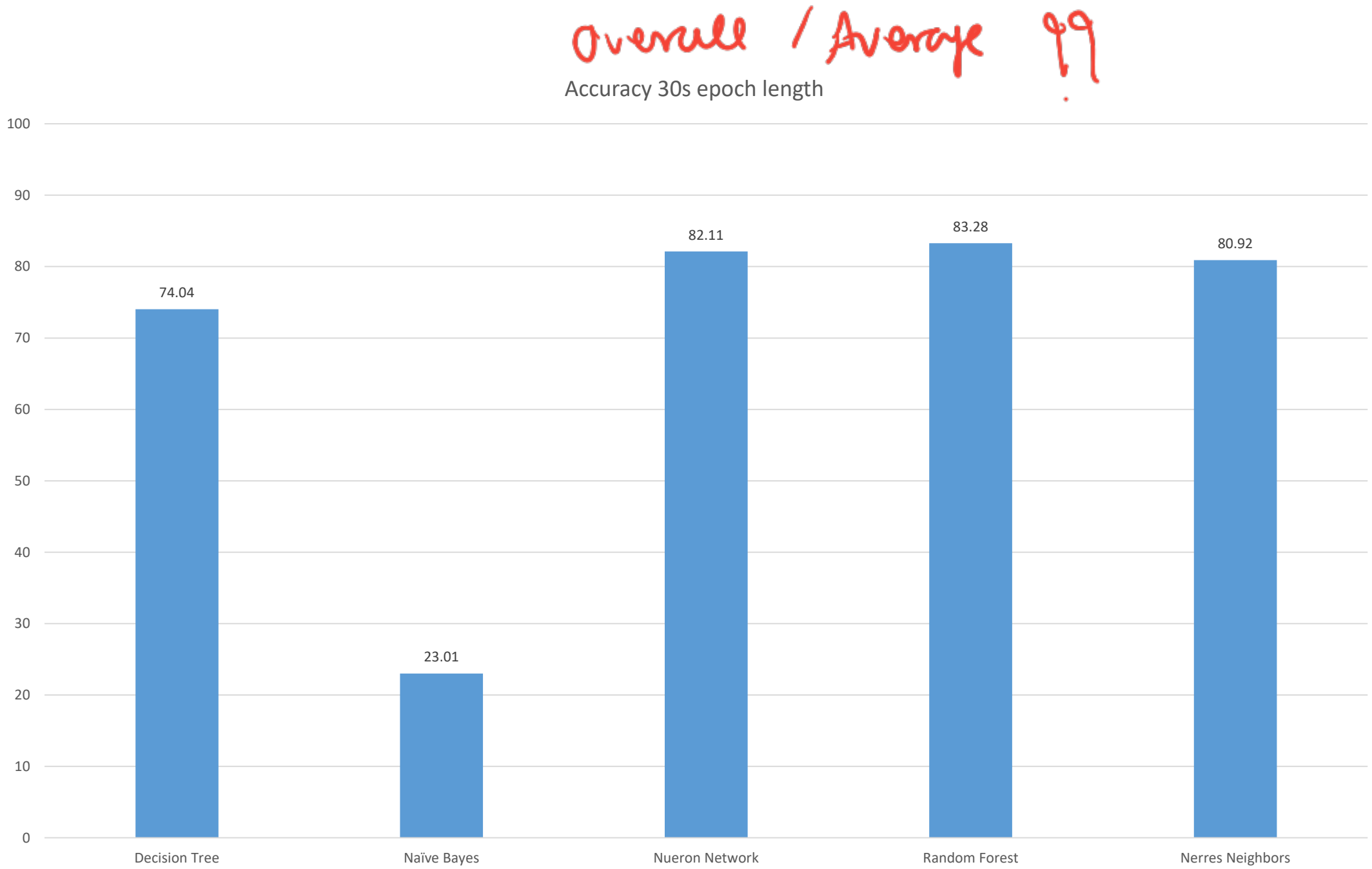
Why is K-Folds Cross Validation?

- the model learns or describes the "noise" in the training data instead of the actual relationships between variables in the data. This noise, obviously, isn't part in of any new dataset, and cannot be applied to it.
- **Overfitting** means that model we trained has trained "too well" and is now, well, fit too closely to the training dataset
- **Underfitting** means that the model does not fit the training data and therefore misses the trends in the data. It also means the model cannot be generalized to new data.

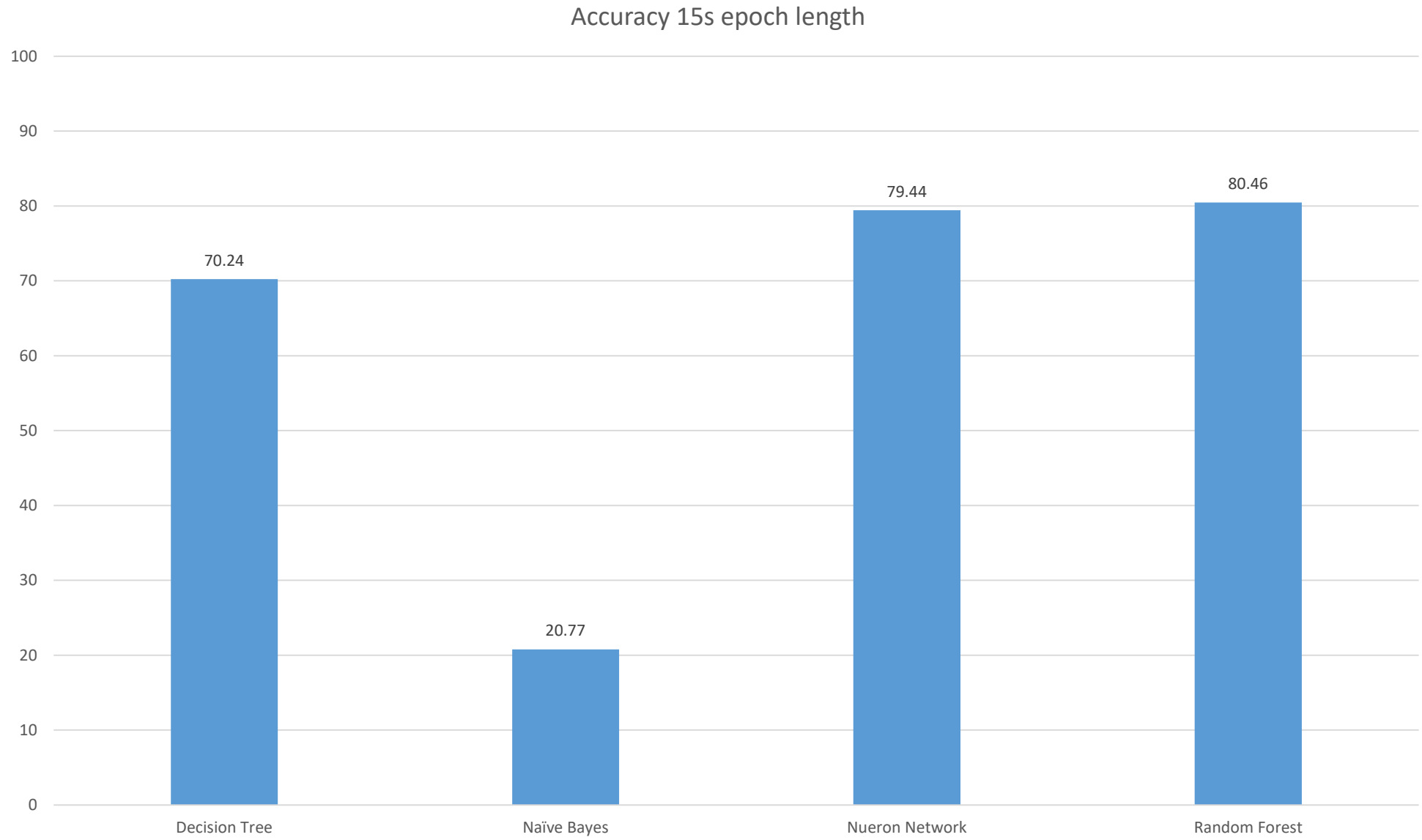
Why ten?
– Extensive experiments have shown that this is the best choice to get
an accurate estimate
– There is also some theoretical evidence for this

Result

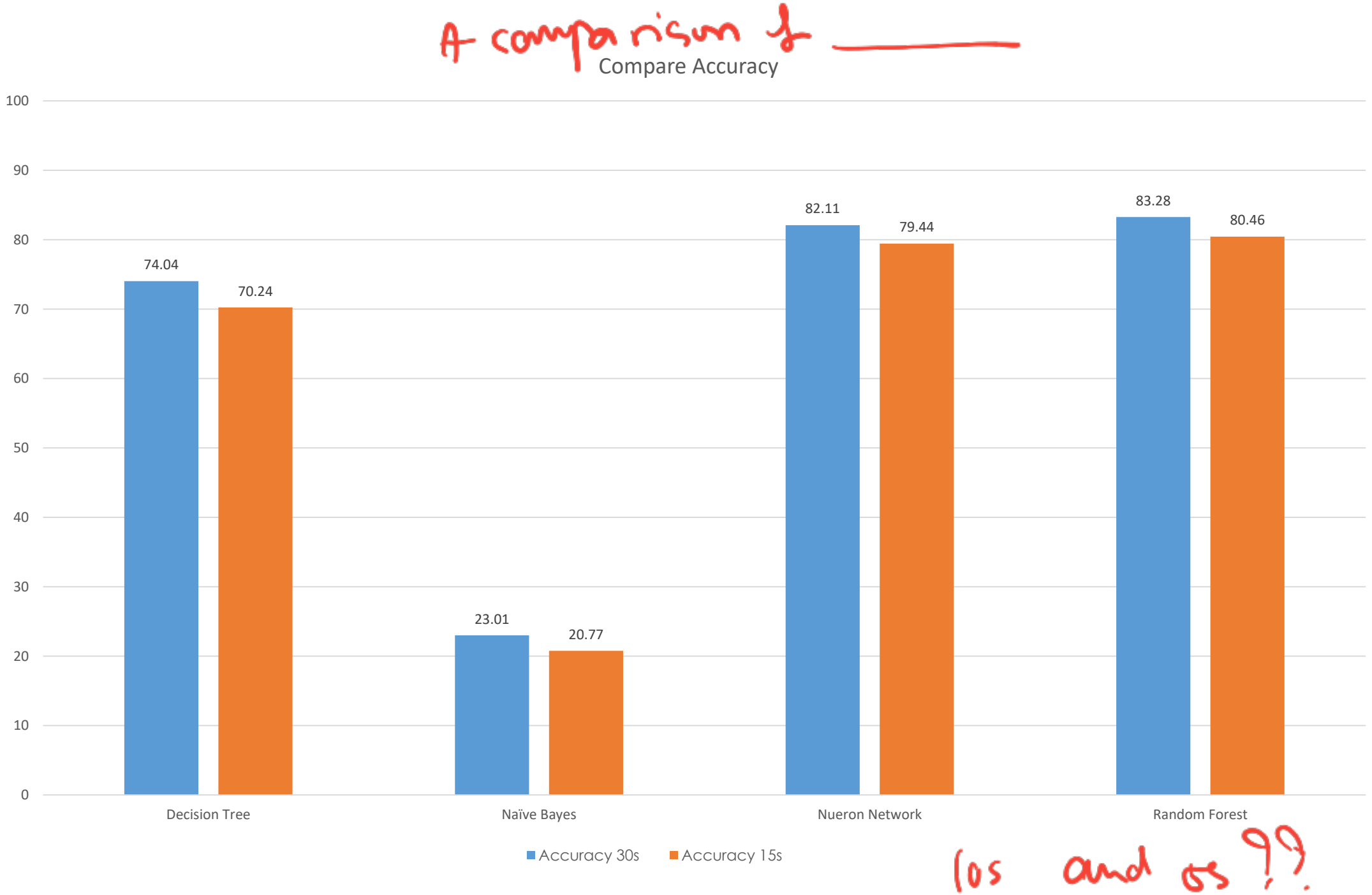
Overall / Average 99



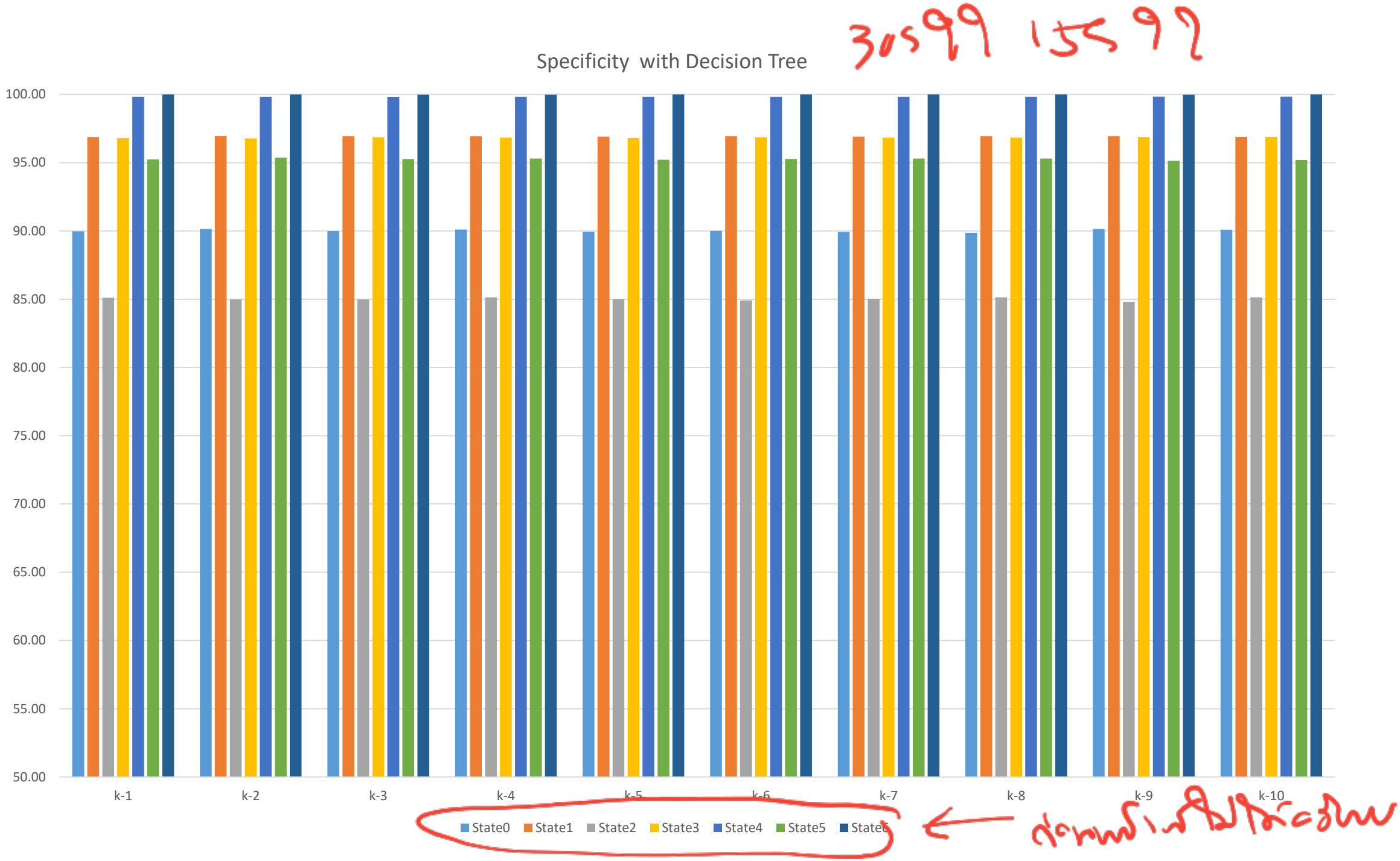
Overall / Averge %



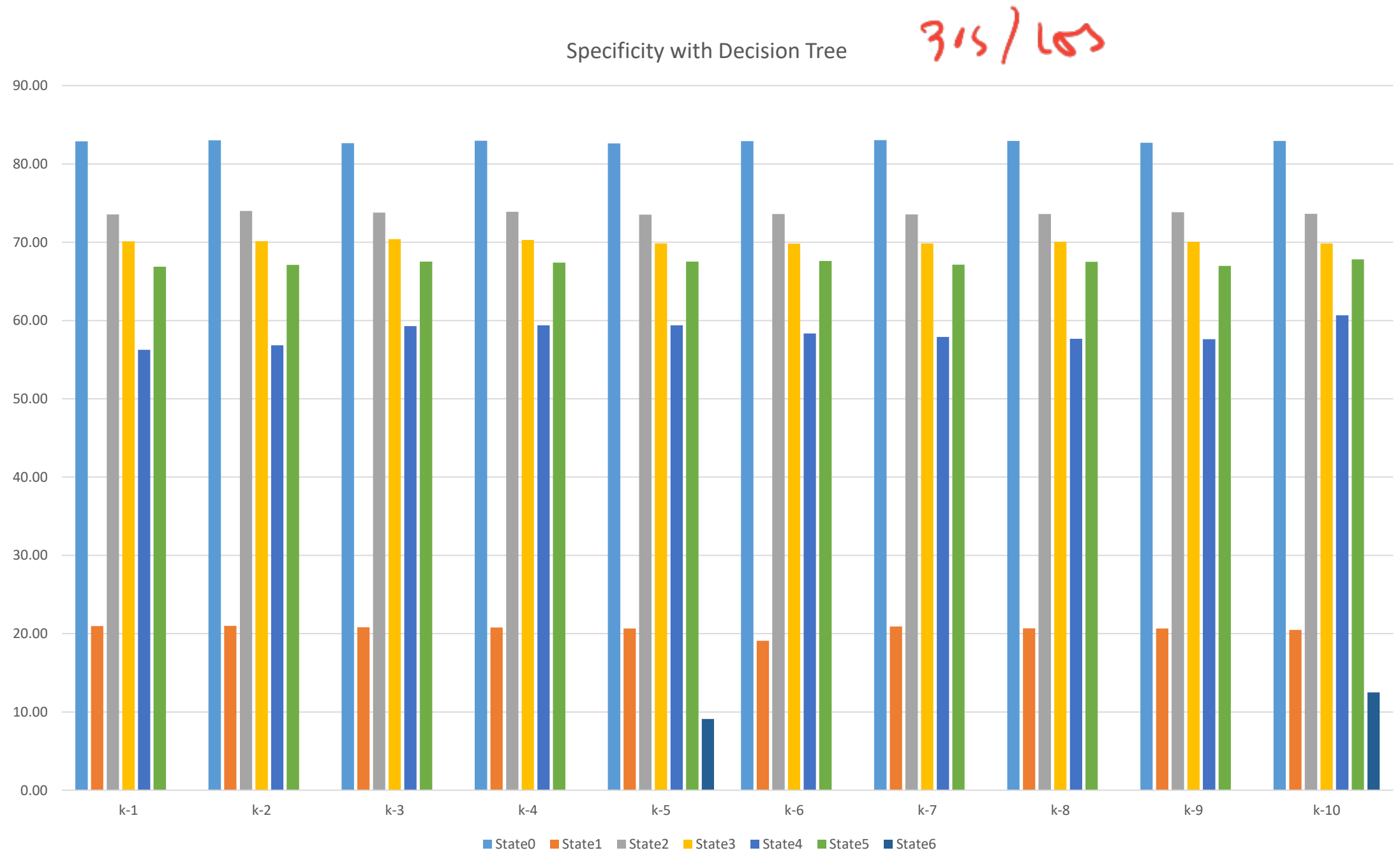
A comparison of ———

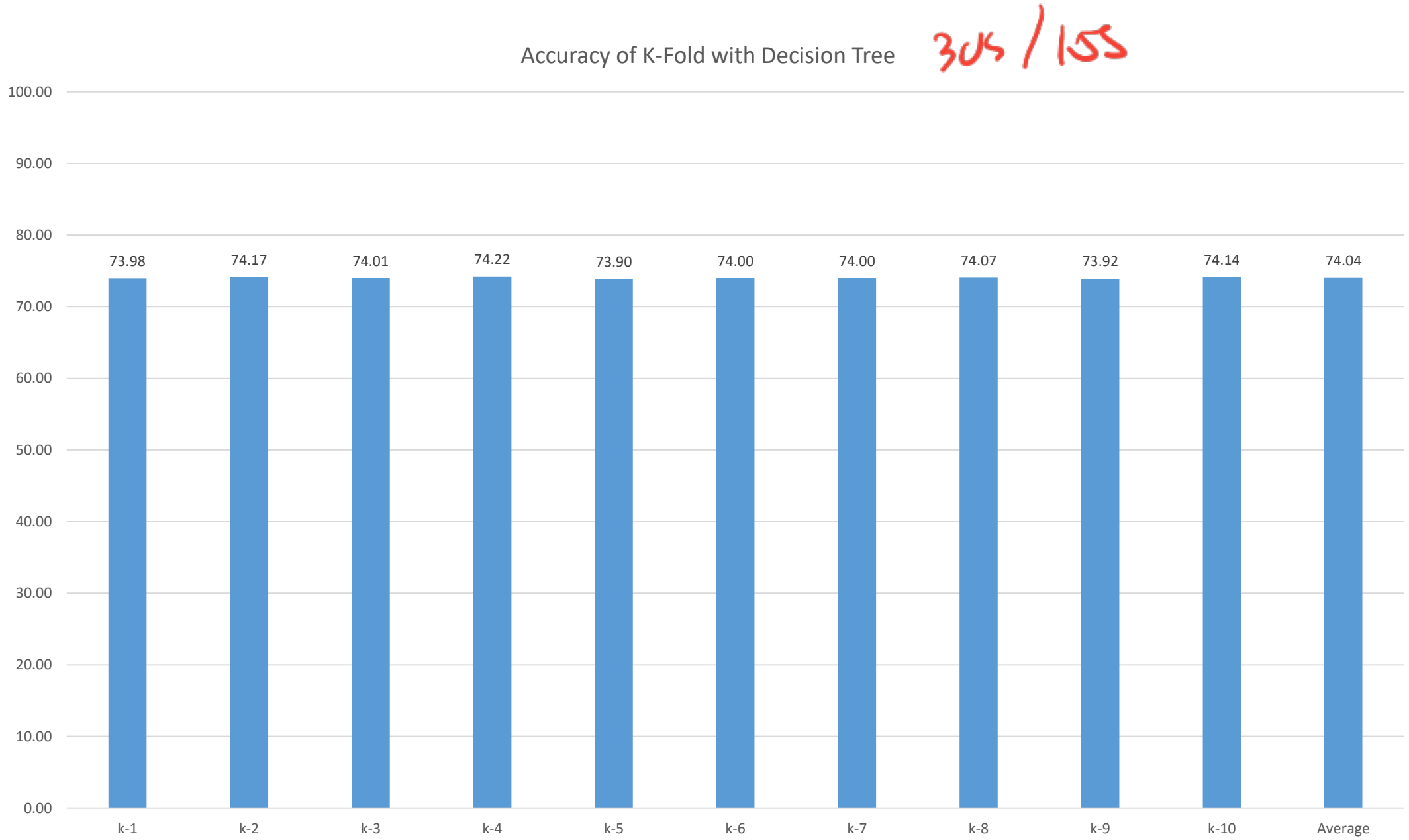


10s and 05 ??


Specificity with Decision Tree
100.00
95.00
90.00
85.00
80.00
75.00
70.00
65.00
60.00
55.00
50.00
k-1
k-2
k-3
k-4
k-5
k-6
k-7
k-8
k-9
k-10
State0
State1
State2
State3
State4
State5


F-Maesure with Decision Tree
100.00
90.00
80.00
70.00
60.00
50.00
40.00
30.00
20.00
10.00
0.00
k-1
k-2
k-3
k-4
k-5
k-6
k-7
k-8
k-9
k-10
State0
State1
State2
State3
State4
State5


Specificity with Decision Tree
90.00
80.00
70.00
60.00
50.00
40.00
30.00
20.00
10.00
0.00
k-1
k-2
k-3
k-4
k-5
k-6
k-7
k-8
k-9
k-10
State0
State1
State2
State3
State4
State5
State6


Accuracy of K-Fold with Decision Tree
305 / 155
100.00
90.00
80.00
70.00
60.00
50.00
40.00
30.00
20.00
10.00
0.00
73.98
74.17
74.01
74.22
73.90
74.00
74.00
74.07
73.92
74.14
74.04
k-1
k-2
k-3
k-4
k-5
k-6
k-7
k-8
k-9
k-10
Average

แนวคิดและวิธีการในการพัฒนางานส่วนที่เหลือ

- Analysis with majority voting
- Evaluate model

Progressing works
Ongoing work

Majority Voting