Subject:
Human activities Recognition using smartphone data

Course: Machine Learning

Student: George Papadopoulos

Human activities Recognition using smartphone data - Introduction

Activities: *running*, *walking*, *cycling.*

End-to-end machine learning system with all phases.

Models: SVM, kNN classifiers.

Emphasis to data preparation, data exploration, feature extraction.

Use of smartphone accelerometer to collect data.

X, Y, Z axes

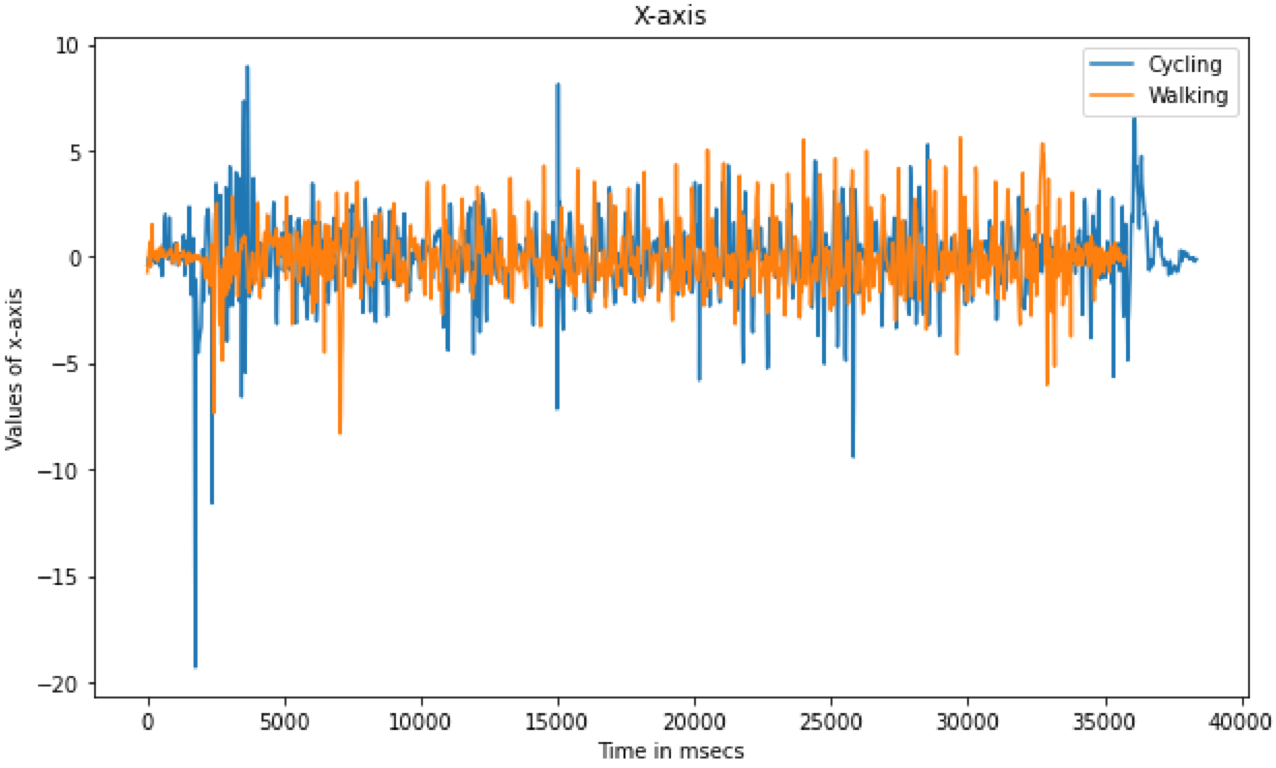
Get/Prepare Phase

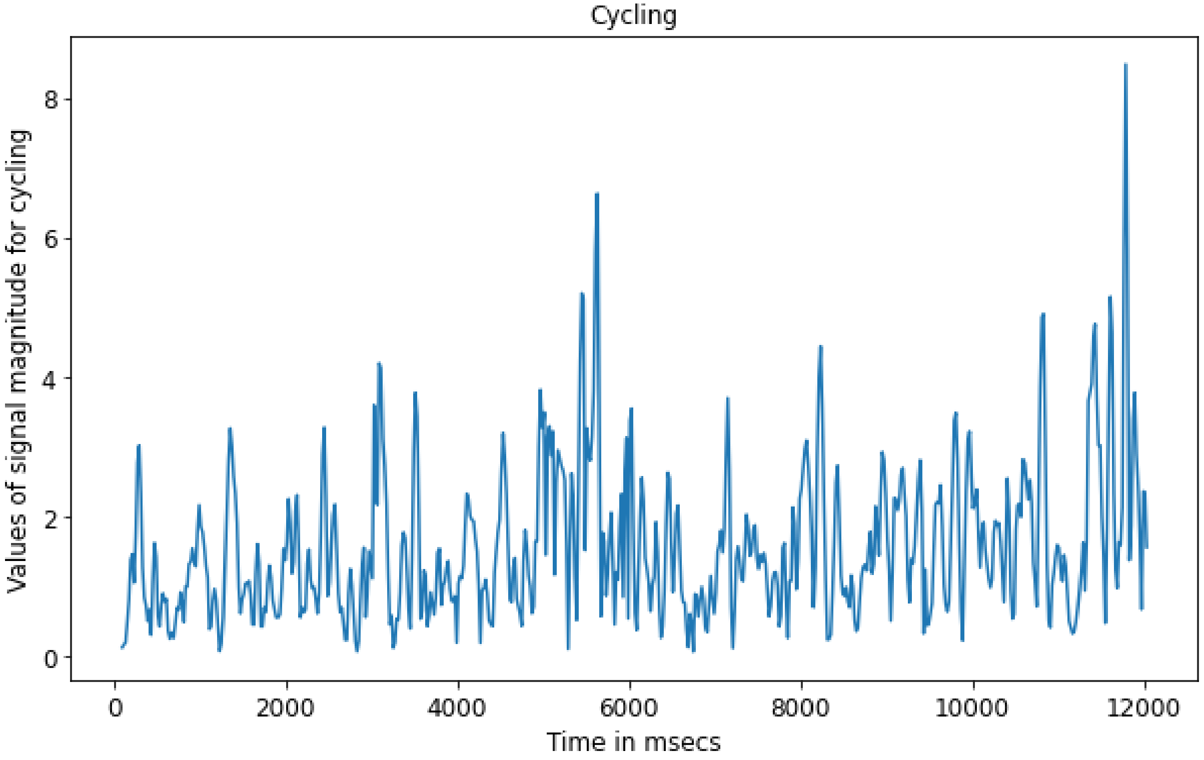
- 6 subjects-persons, 10 samples per activity, 180 samples totally.
    - Balanced dataset.
    - 4 subjects train-validation set, 2 subjects test set.
- Each sample 30+ secs. 0-10 preparation, 10-22 clear sample, 22-30+ stop.
- Use of the same smartphone to avoid differences of sensors.
- Sampling at 50 Hz. Measurement unit: m/s^2.
- Annotation after sampling.
- Upload to GitHub.
- Parser creation to read .txt samples files.

Prepare – Exploration phase

- Cutting of X, Y, Z signals and keep only 12 secs.

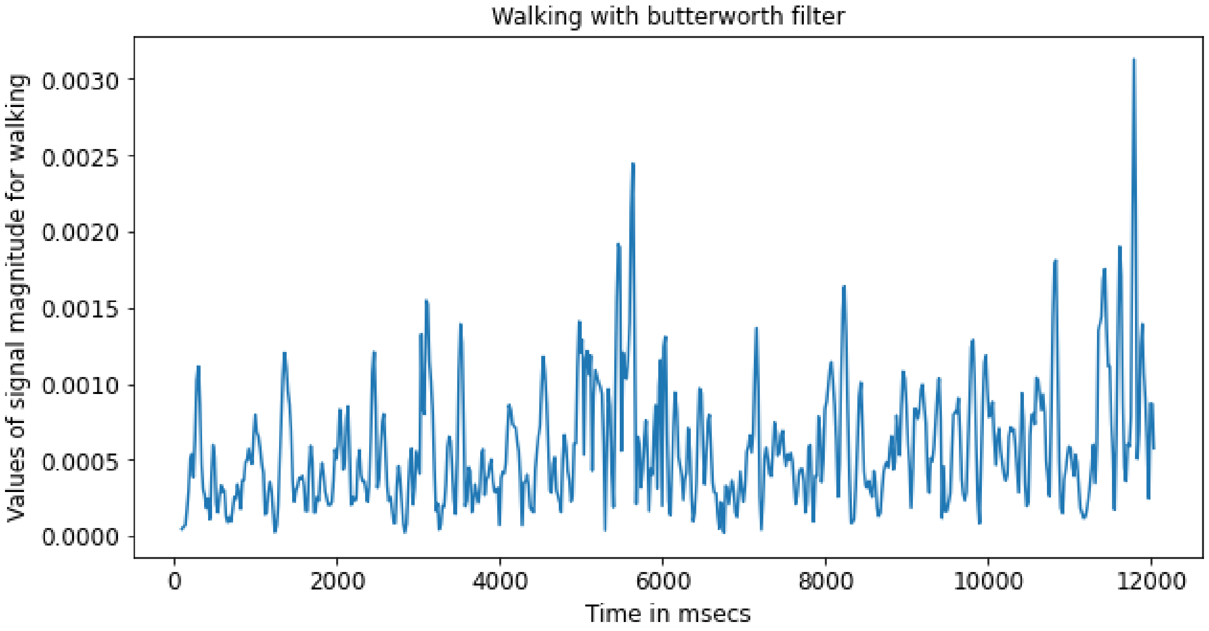
- Problem with orientation changes of smartphone.
- Computation of X-Y magnitude to solve it.

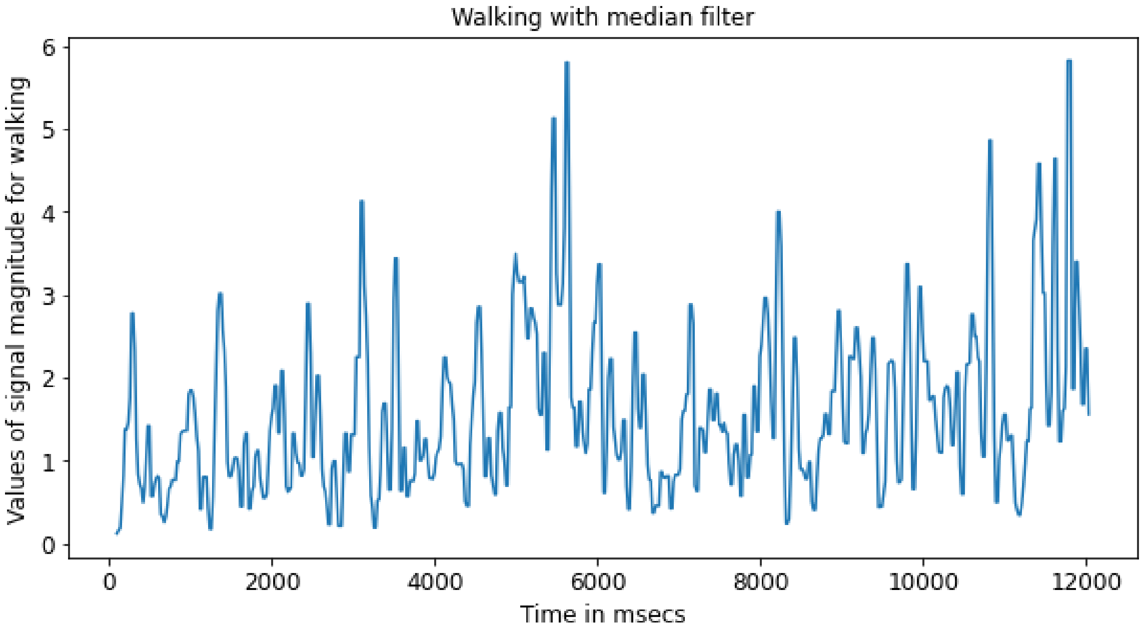




Preparation phase - Denoising

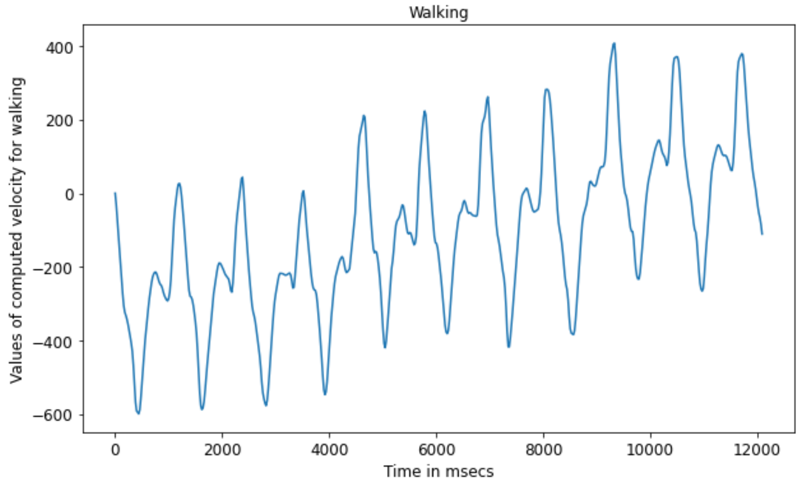
- Experimentation with Butterworth low pass filter and Median filter.
- Choose Median filter which smooths the signal.

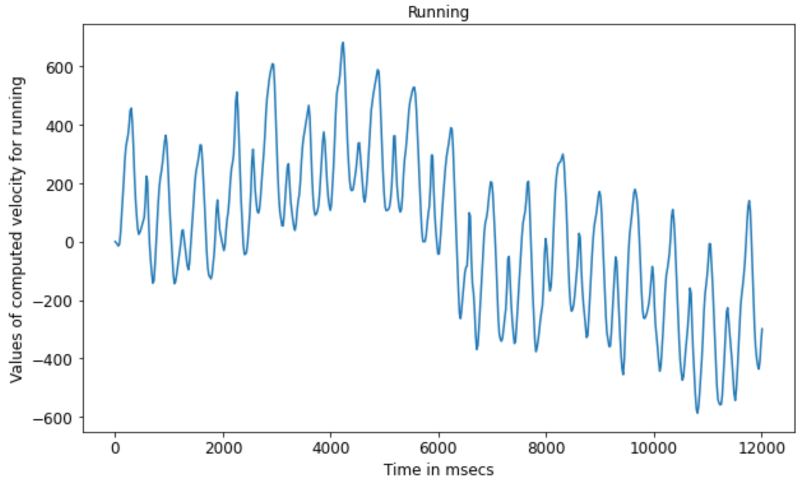




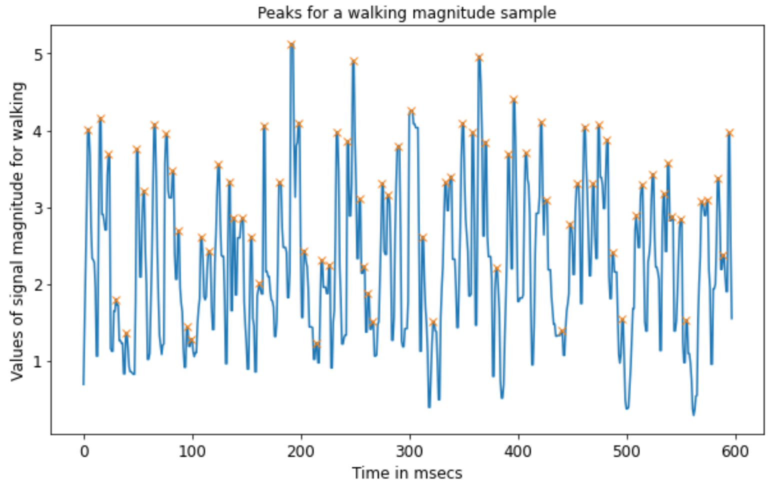
Exploration phase – Discover patterns

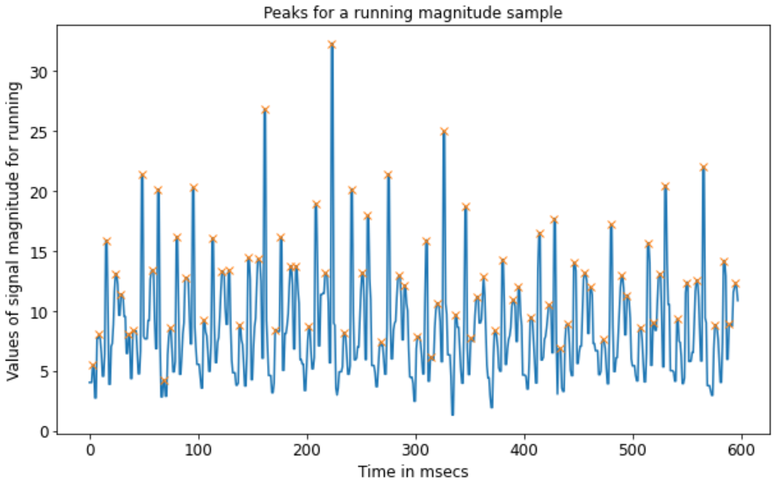
- *Velocity* considered significant to find patterns.
- Unfortunately, depends on initial velocity.
- But *dominant frequency* can be used.





- *Peaks* of signals seems to have different values and *widths* between them.



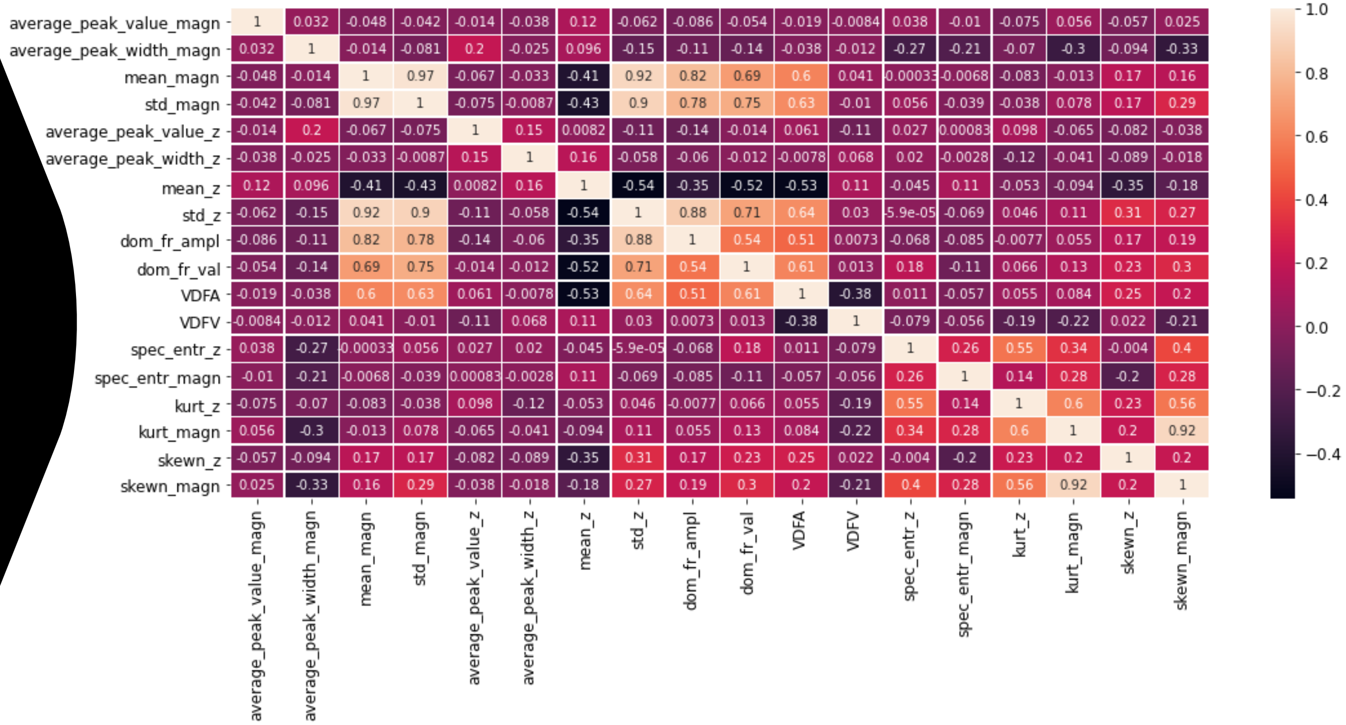


Exploration phase – Feature examination

- Possible features:
  - Statistics of Z-axis signal and X-Y magnitude: *mean, standard deviation, kurtosis, skewness.*
  - *Average peak values, average peak widths.*
  - *Spectral Entropy.*
  - *Dominant Frequency value* and *amplitude* for Z-axis, X-Y magnitude and velocity using Fourier Transformation.
- Examine correlations using *correlation matrix.*
  - Highly correlated:

mean_magn – std_mean – std_z

kurt_magn – skew_magn

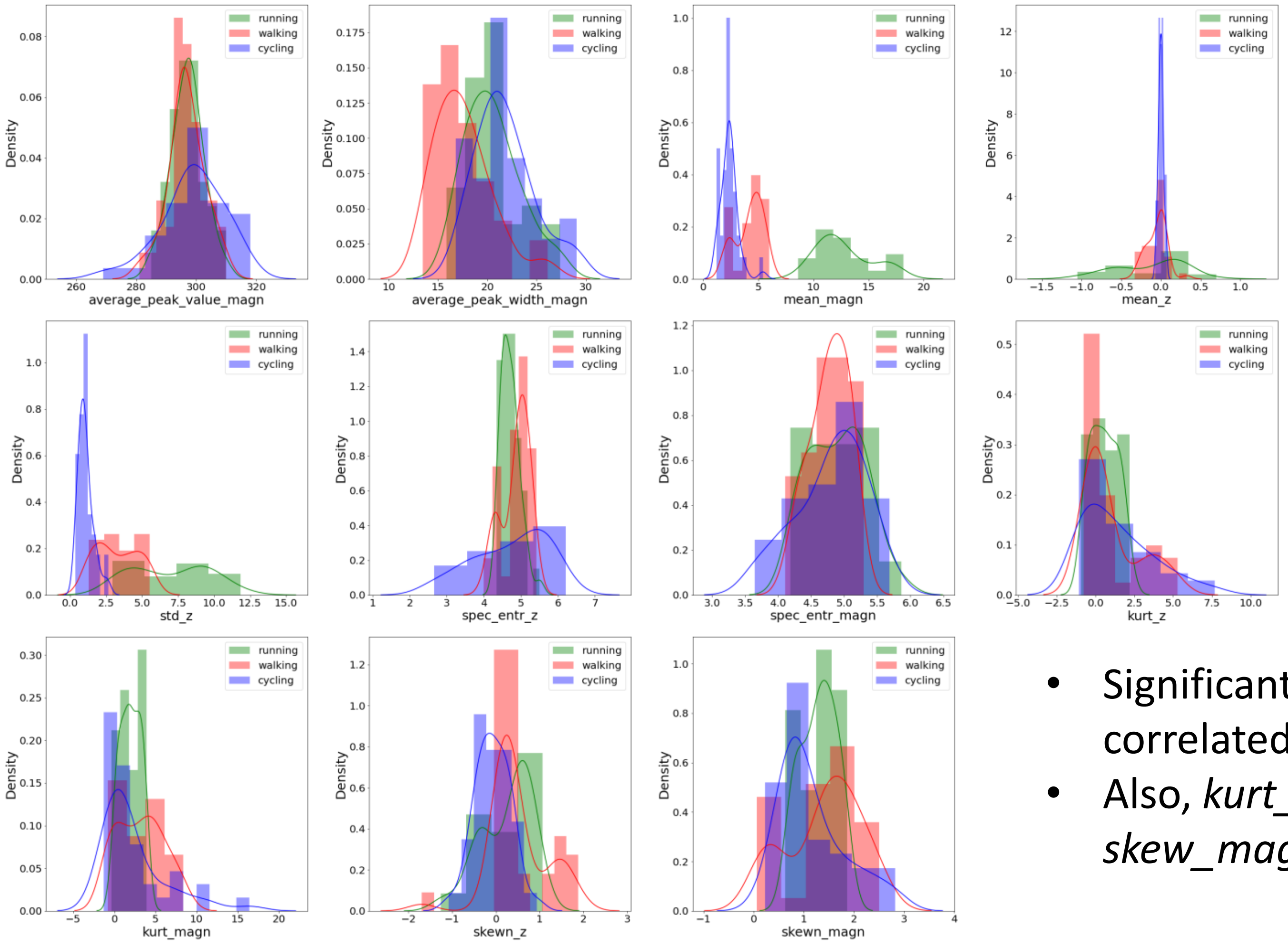


	average_peak_value_magn	average_peak_width_magn	mean_magn	std_magn	average_peak_value_z	average_peak_width_z	mean_z	std_z	dom_fr_ampl	dom_fr_val	VDFA	VDFV	spec_entr_z	spec_entr_magn	kurt_z	kurt_magn	skewn_z	skewn_magn
average_peak_value_magn	1	0.032	-0.048	-0.042	-0.014	-0.038	0.12	-0.062	-0.086	-0.054	-0.019	-0.0084	0.038	-0.01	-0.075	0.056	-0.057	0.025
average_peak_width_magn	0.032	1	-0.014	-0.081	0.2	-0.025	0.096	-0.15	-0.11	-0.14	-0.038	-0.012	-0.27	-0.21	-0.07	-0.3	-0.094	-0.33
mean_magn	-0.048	-0.014	1	0.97	-0.067	-0.033	-0.41	0.92	0.82	0.69	0.6	0.041	-0.00033	-0.0068	-0.083	-0.013	0.17	0.16
std_magn	-0.042	-0.081	0.97	1	-0.075	-0.0087	-0.43	0.9	0.78	0.75	0.63	-0.01	0.056	-0.039	-0.038	0.078	0.17	0.29
average_peak_value_z	-0.014	0.2	-0.067	-0.075	1	0.15	0.0082	-0.11	-0.14	-0.014	0.061	-0.11	0.027	0.00083	0.098	-0.065	-0.082	-0.038
average_peak_width_z	-0.038	-0.025	-0.033	-0.0087	0.15	1	0.16	-0.058	-0.06	-0.012	-0.0078	0.068	0.02	-0.0028	-0.12	-0.041	-0.089	-0.018
mean_z	0.12	0.096	-0.41	-0.43	0.0082	0.16	1	-0.54	-0.35	-0.52	-0.53	0.11	-0.045	0.11	-0.053	-0.094	-0.35	-0.18
std_z	-0.062	-0.15	0.92	0.9	-0.11	-0.058	-0.54	1	0.88	0.71	0.64	0.03	-5.9e-05	-0.069	0.046	0.11	0.31	0.27
dom_fr_ampl	-0.086	-0.11	0.82	0.78	-0.14	-0.06	-0.35	0.88	1	0.54	0.51	0.0073	-0.068	-0.085	-0.0077	0.055	0.17	0.19
dom_fr_val	-0.054	-0.14	0.69	0.75	-0.014	-0.012	-0.52	0.71	0.54	1	0.61	0.013	0.18	-0.11	0.066	0.13	0.23	0.3
VDFA	-0.019	-0.038	0.6	0.63	0.061	-0.0078	-0.53	0.64	0.51	0.61	1	-0.38	0.011	-0.057	0.055	0.084	0.25	0.2
VDFV	-0.0084	-0.012	0.041	-0.01	-0.11	0.068	0.11	0.03	0.0073	0.013	-0.38	1	-0.079	-0.056	-0.19	-0.22	0.022	-0.21
spec_entr_z	0.038	-0.27	-0.00033	0.056	0.027	0.02	-0.045	-5.9e-05	-0.068	0.18	0.011	-0.079	1	0.26	0.55	0.34	-0.004	0.4
spec_entr_magn	-0.01	-0.21	-0.0068	-0.039	0.00083	-0.0028	0.11	-0.069	-0.085	-0.11	-0.057	-0.056	0.26	1	0.14	0.28	-0.2	0.28
kurt_z	-0.075	-0.07	-0.083	-0.038	0.098	-0.12	-0.053	0.046	-0.0077	0.066	0.055	-0.19	0.55	0.14	1	0.6	0.23	0.56
kurt_magn	0.056	-0.3	-0.013	0.078	-0.065	-0.041	-0.094	0.11	0.055	0.13	0.084	-0.22	0.34	0.28	0.6	1	0.2	0.92
skewn_z	-0.057	-0.094	0.17	0.17	-0.082	-0.089	-0.35	0.31	0.17	0.23	0.25	0.022	-0.004	-0.2	0.23	0.2	1	0.2
skewn_magn	0.025	-0.33	0.16	0.29	-0.038	-0.018	-0.18	0.27	0.19	0.3	0.2	-0.21	0.4	0.28	0.56	0.92	0.2	1

Feature selection

- Selection using distribution plots, backward Elimination, SelectKBest

Activity analysis for the selected features

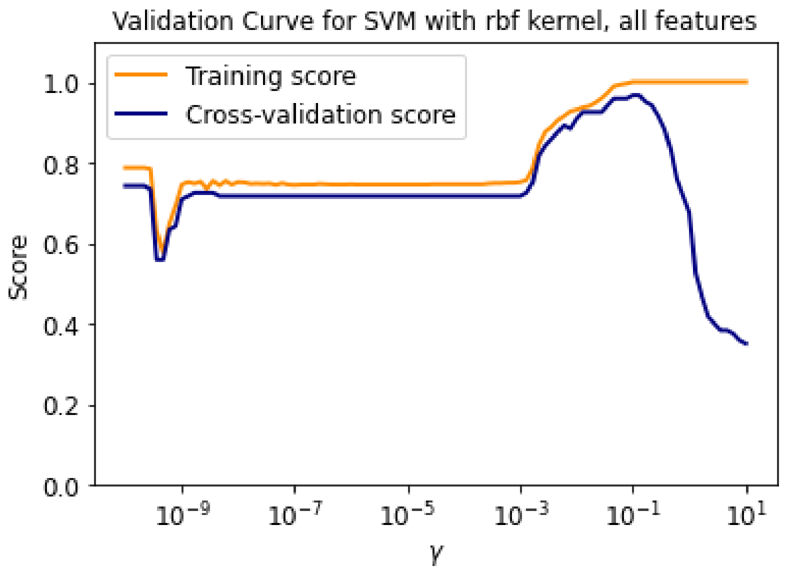


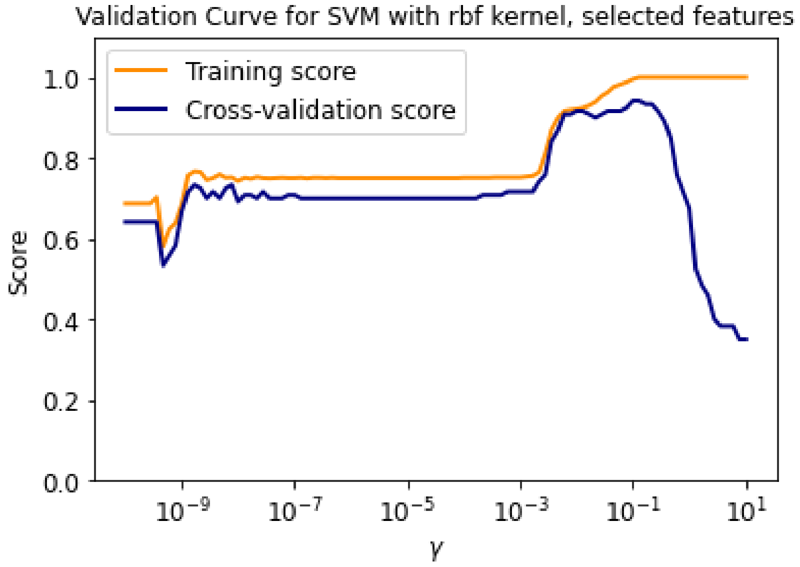
```
                    Feature_Name       Score
2                      mean_magn  446.725187
3                       std_magn  179.601351
7                          std_z  112.165381
8                    dom_fr_ampl   70.188456
9                     dom_fr_val   24.900838
1        average_peak_width_magn   22.306325
10                          VDFA   17.493667
16                       skewn_z   10.604691
6                         mean_z    2.647377
14                        kurt_z    2.422728
15                     kurt_magn    1.835124
11                          VDFV    1.784572
4          average_peak_value_z    1.590847
17                    skewn_magn    1.431152
0        average_peak_value_magn    0.917507
13                spec_entr_magn    0.806367
```

- Significant *mean_magn, std_magn* and *std_z,* but strong correlated. *mean_magn* more important, exclude other two.
- Also, *kurt_magn* and *skew_magn* correlated, exclude *skew_magn.*

Model phase – SVM train

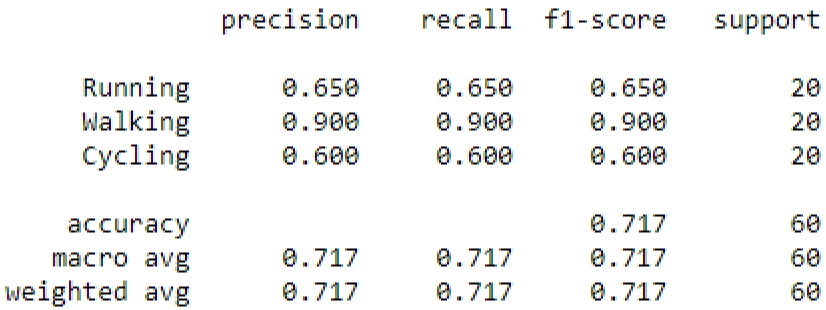
- Validation curves for two dataset with selected features and all features, for different *gamma* parameters.
  - Examine where model begins overfitting and which dataset is best.
  - Use of *Cross Validation*, *pipeline* and *StandardScaler.*
- Use of *GridSearchCV* to determine the best values of parameters *gamma* and *C* for *rbf kernel.*
  - The same for *polynomial kernel* with extra parameter *degree.*
- Score: *mean accuracy* of subclasses.
- Use of *cross_validation_score* to compare two kernels.
- Rbf: 94%, Polynomial: *87%.*


Validation Curve for SVM with rbf kernel, all features

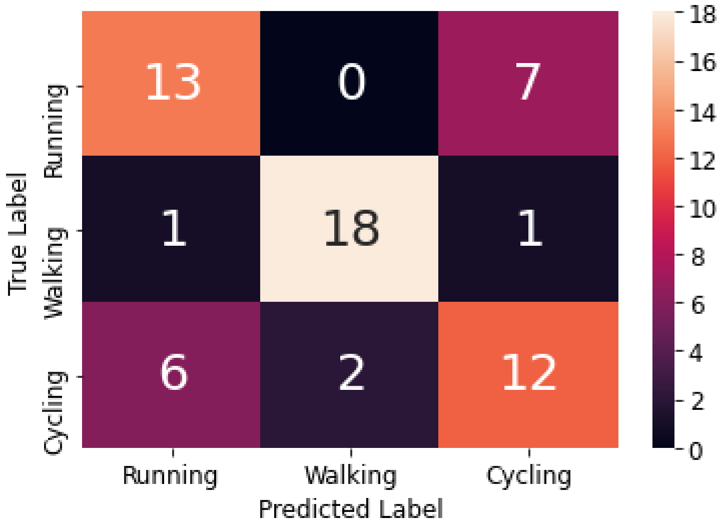

Validation Curve for SVM with rbf kernel, selected features

Model phase – SVM test

- Use of test set for first time to test SVM performance.
- Creation of classification report and confusion matrix.



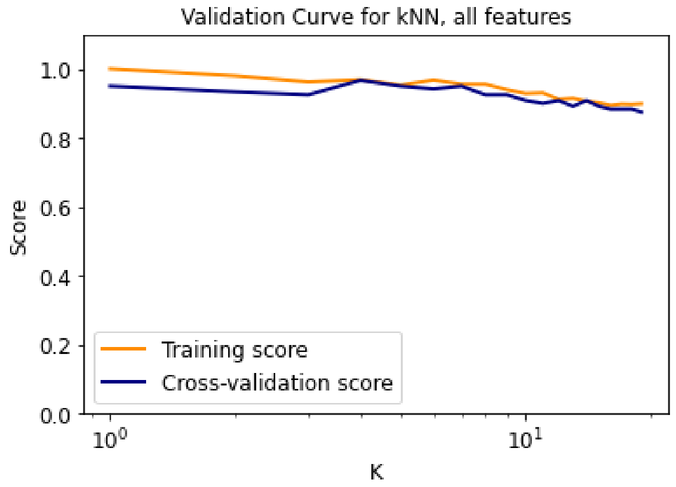
```
              precision    recall  f1-score   support

     Running      0.650     0.650     0.650        20
     Walking      0.900     0.900     0.900        20
     Cycling      0.600     0.600     0.600        20

    accuracy                          0.717        60
   macro avg      0.717     0.717     0.717        60
weighted avg      0.717     0.717     0.717        60
```

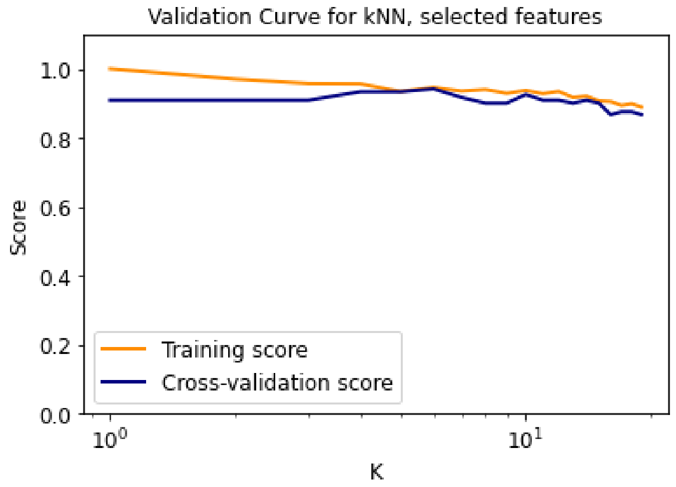


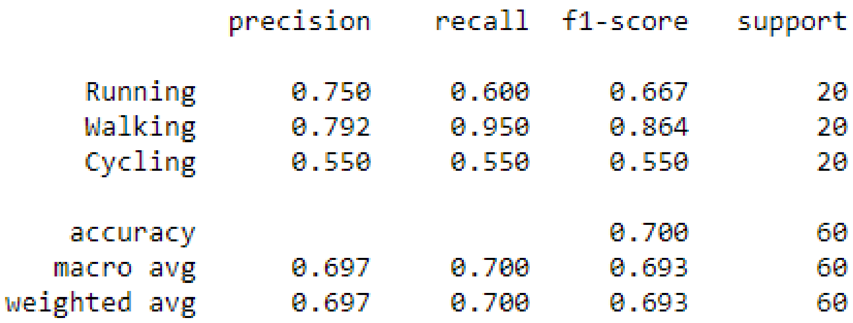
- Scale test set separately.
- Unexpected results, walking class has the least misclassified samples and so the highest scores.
- Cycling has the lowest scores.
- Classifier is confused between running and cycling.
- Interesting, symmetrical precision-recall.
- Overall f1 71%. Great difference between train-validation and test scores. More samples for training might be needed.

Model phase – kNN train and test

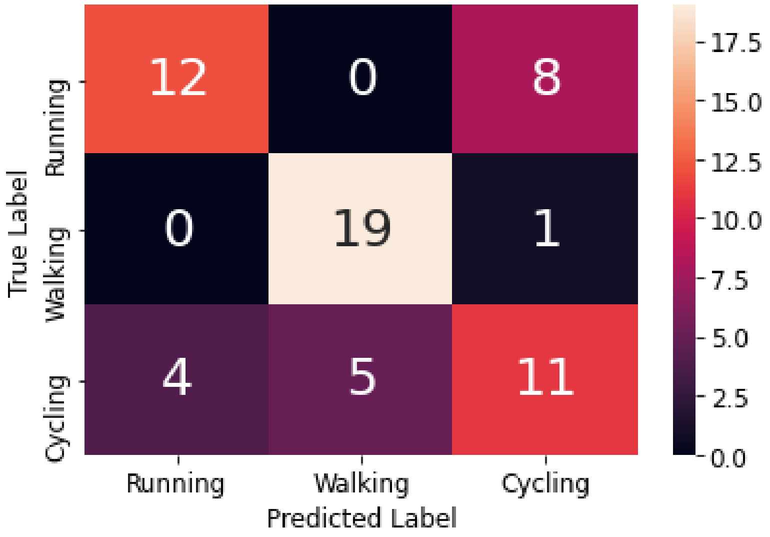
- Slight discrepancies between two datasets for different values of parameter *K-nearest neighbors*.
- Determine best value with *GridSearchCV* as before.







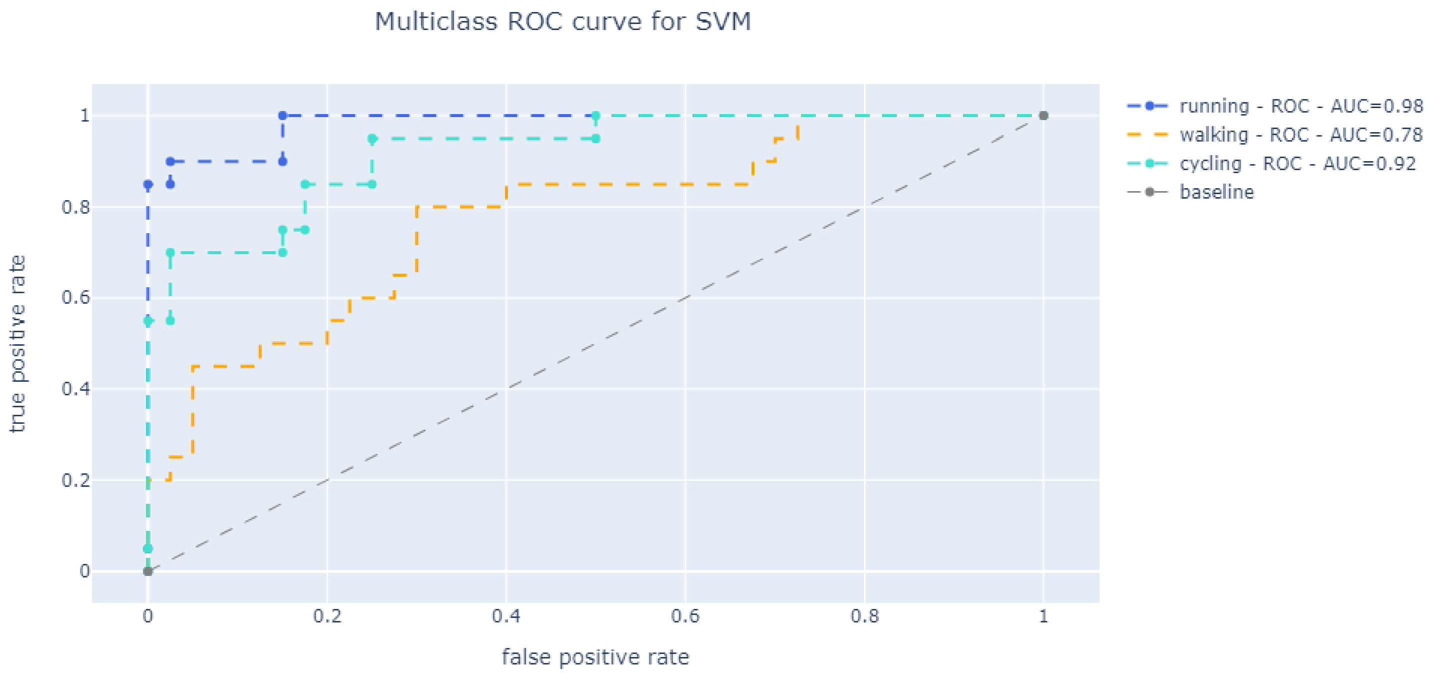
```
              precision    recall  f1-score   support

     Running      0.750     0.600     0.667        20
     Walking      0.792     0.950     0.864        20
     Cycling      0.550     0.550     0.550        20

    accuracy                          0.700        60
   macro avg      0.697     0.700     0.693        60
weighted avg      0.697     0.700     0.693        60
```

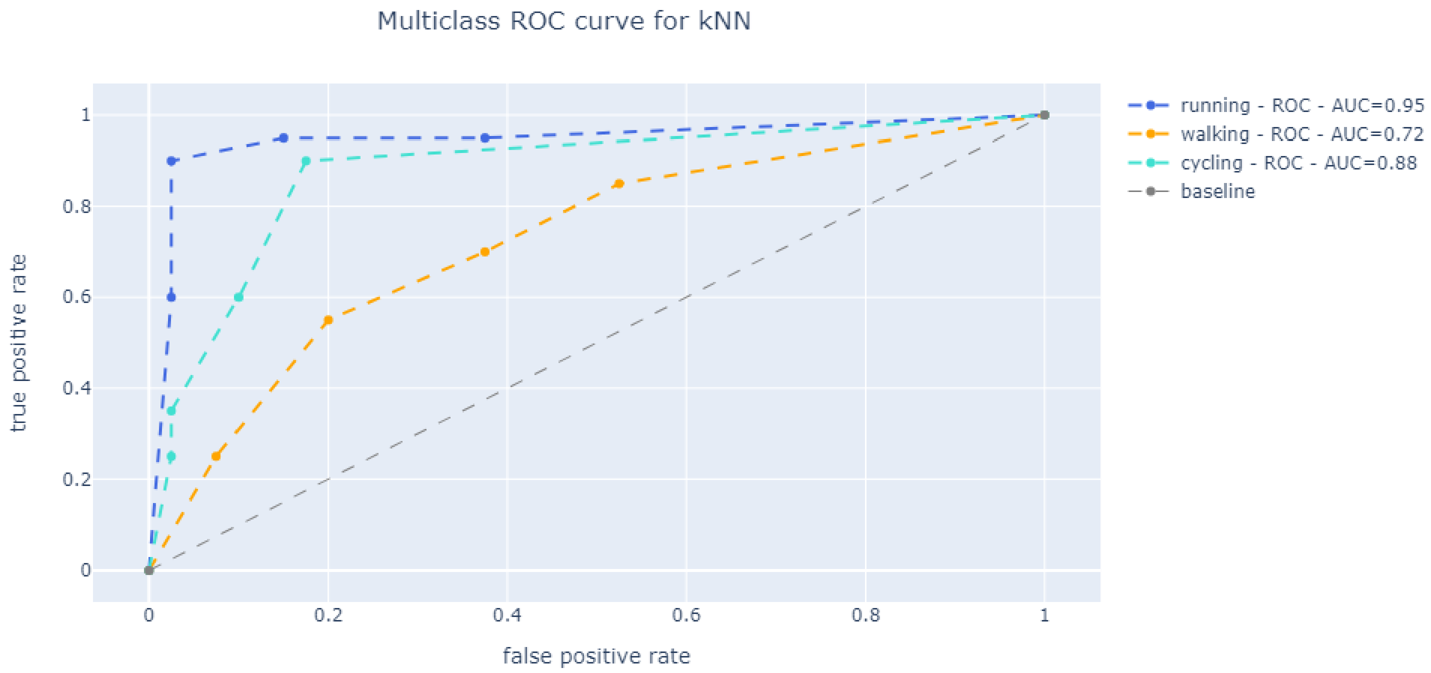


- Same rank as SVM, but different scores.
- Cycling almost equal scores to random classification.
- Walking high recall due to only 1 FN, lower precision due to 5 FP.
- Running, walking overlapping cycling
- Overall f1 70%, almost equal to SVM.

Models' comparison – Communicate results

- ROC curves: Each class against the others.
- From ROC curves can be concluded SVM has greater performance for all classes.
- Highest SVM ROC-AUC score : 98% for running. Meaning from 100 samples predicted as *running*, 2 will not belong to *running* class.
- Highest kNN: 95% for *running*.
- Concluding: Using smartphone in trouser pocket, accurate enough to monitor activities especially *running* and *cycling*. Building of a simple app to use by athletes or someone who cares about daily activity performances and times.
- Future work: Improvement of *walking* class predictions using more training data, longer duration. Or/And time-window based model.





Thank you for your attention!

Question Time..