Group 5: Gahyun Kim, Thai Nguyen, Colin Stern

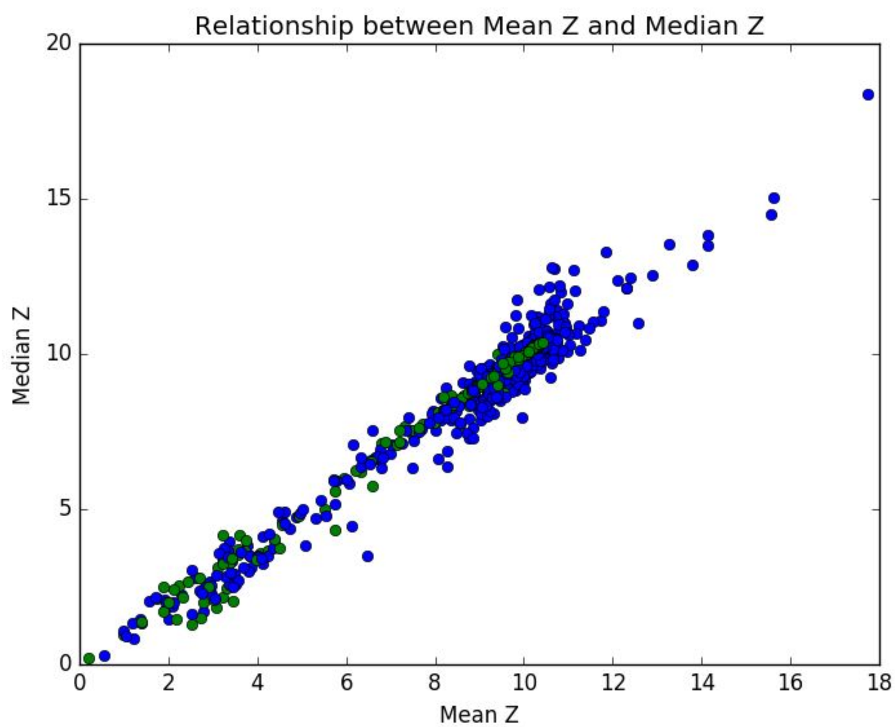
A2 part B: Supervised Classification

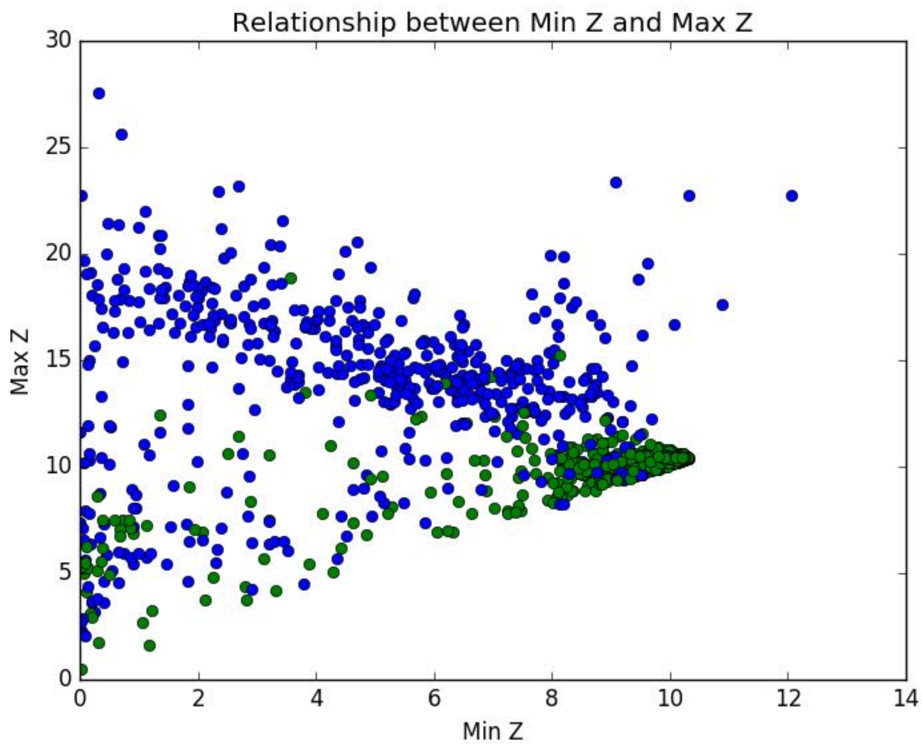
PART 2: Feature Extraction

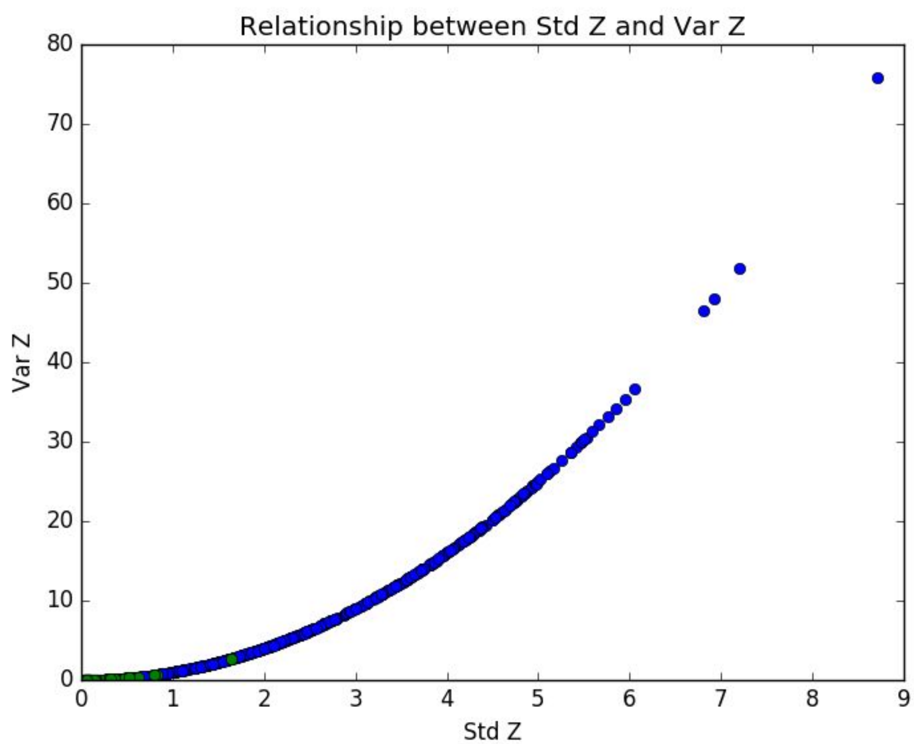
1. From at least 3 of the categories of features above, extract 2 such features. These functions should be included in `features.py.`

   - Statistical features: mean, median, min, max, standard deviation, variance
   - Statistical features of magnitudes $\sqrt{x^2+y^2+z^2}$ (mean, median, min, max, standard deviation, variance)
   - Entropy of histogram of 5 bins
   - Length of feature vector = 6x3 (statistical features) + 6 (magnitude statistical features) + 1 (entropy) = 25

2. Plot at least 3 graphs where both the `x` and `y` axes corresponds to a different feature and each point is colored according to its label. Refer to the plotting code from the script in part 1 of the assignment.




Relationship between Min Z and Max Z
Max Z
Min Z
0
5
10
15
20
25
30
0
2
4
6
8
10
12
14


Relationship between Std Z and Var Z
Var Z
Std Z
0
10
20
30
40
50
60
70
80
0
1
2
3
4
5
6
7
8
9

3. Describe based on the visualizations which features appear to work best and which discriminate activities poorly.

    - Comparing the distribution of data in the three plots, we see that plotting with standard deviation Z and variance Z as the axes works best in discriminating the activities. Since the standard deviation Z and variance Z works best in discriminating activities, we know that the sample data fits a normal distribution shape. We can see in this plot that as standard deviation increases, so does variance.We also see a high correlation between the two variables. Thus, we can conclude that the data was most likely generated with normal distribution.

    - The min Z and max Z does an okay job at discriminating activities. We can two separate clusters in the plot (blue on top, green on bottom). We can try to discriminate the two activities in the min Z and max Z plot by drawing a horizontal line. Although, we get most of one class above the line, there is still a lot of entropy in the bottom half of the line. Thus, the min Z and max Z plots are not good for discriminating the activities.

    - Lastly, the mean Z and median Z does the poorest job at discriminating activities. It is difficult to differentiate the activities due to its overall high entropy.

PART 3: Training a Classifier

1. Train at least **four** decision tree classifiers varying the `max_depth` and `max_features` parameters. Evaluate each decision tree classifier and report the accuracy, precision and recall metrics averaged over 10-fold CV.

```
train_and_predict(
    "Decision Tree Max Depth 5",
    DecisionTreeClassifier(criterion="entropy", max_depth=5))
train_and_predict(
    "Decision Tree Max Depth 10",
    DecisionTreeClassifier(criterion="entropy", max_depth=10))
train_and_predict(
    "Decision Tree Max Features 5",
    DecisionTreeClassifier(criterion="entropy", max_features=5))
train_and_predict(
    "Decision Tree Max Features 10",
    DecisionTreeClassifier(criterion="entropy", max_features=10))
```

2. Include the Graphviz image displaying at least two of these trees. Speak briefly to the visualization, comparing the trees.
   <See attached images: “Decision Tree Max Depth 5.png” and “Decision Tree Max Features 5.png”
   - The most striking aspect of the visualization of the trees based on max features is their size: The tree built with max_features=5 is far larger than the tree built with max_depth=5; the same can be said between max_features=10 and max_depth=10. More generally, as the depth and features increase, the size of the tree grows exponentially.
   - In addition, all of the features are not used at every division of the data. In the tree with max_depth=5, the first divisions are on the standard deviation of the Z-axis and the mean of Z. After the second division, the magnitude, max of X and min of X is used for splitting data more often.

3. Describe what effect each parameter has on your results. Why do you think that is?
   - Increasing the number of features exponentially increases the complexity - after more than 10 features, the size of the tree is almost unmanageable. Increasing the depth of the tree allows for more accuracy because data can be divided more precisely: Labels can be assigned to smaller sets of data points.
   - In addition, increasing the depth of the tree does not results in greater accuracy and precision. This this probably because that we don’t have much training data and a tree with depth 5 is already complex to learn about the pattern in the data.

4. Is the decision tree classifier a linear or nonlinear classifier? Explain.
    - It is nonlinear because the data is split on multiple features. While each individual split is linear, the entire tree is not. The result is not polynomial, but simply nonlinear.

5. Train at least one other model and report the cross-validated accuracy, precision and recall metrics on the test data. Does it do better than the decision tree classifier?

```
train_and_predict("Linear SVC", LinearSVC())
```

The Linear SVC we tested has worse accuracy and precision than all of the other decision trees. This is because the data cannot be separated linearly; the decision trees, being nonlinear, are able to label the data more accurately than a linear classifier.

Decision Tree Max Depth 5

- Average accuracy: 0.947490589711
- Average recall: 0.945637098175
- Average precision: 0.936181907124

Decision Tree Max Depth 10

- Average accuracy: 0.94347553325
- Average recall: 0.937542678448
- Average precision: 0.93364819283

Decision Tree Max Features 5

- Average accuracy: 0.937264742785
- Average recall: 0.927899401304
- Average precision: 0.928157967969

Decision Tree Max Features 10

- Average accuracy: 0.940025094103
- Average recall: 0.930365877302
- Average precision: 0.931808475602

Linear SVC

- Average accuracy: 0.835947302384
- Average recall: 0.850843563557
- Average precision: 0.826601644774

PART 4: Completing the Loop

1. Describe the data collection process.

    a. **Research subjects**: Group members Gahyun and Colin.

    b. **Activities**:
        - Sitting
        - Walking
        - Running
        - Jumping

    c. **Performance times** of activities:
        - Sitting, walking, and jumping were performed for five consecutive minutes. Running was performed for seven minutes because there were some lags in data collection.

    d. **Phone orientation**: Held vertically.

    e. **Motion performances**:
        - <u>Sitting</u>: Held phone vertically while sitting and doing minor tasks (writing, talking)
        - <u>Walking</u>: Maintained a normal walking pace and kept gait natural.
        - <u>Running</u>: Jogged briskly around the Computer Science building.
        - <u>Jumping</u>: Jumped in place.

    f. Accounting for variation in the phone position/orientation and the activity style
        - We made sure to keep the phone orientation as consistent between subjects and experiments as possible. Except for minor aberrations, the phone was held vertically while collecting data. The application is designed to work when held in the same orientation as the training data.

2. Which classification algorithm and parameters did you select in your final system? Why?
    - We selected a two Decision Trees with a max depth of 5 and 10 respectively and two Decision Trees with max features of 5 and 10 respectively, and a Linear SVC to compare with the results we got from 2 activities only.

3. Report the accuracy, precision and recall metrics for the classifier and features you decided to use. How do the results compare to the results on the sample data? Briefly speak to how well your algorithm works in practice, drawing on the empirical results. Do they match up?

For precision and recall here, we use macro averaging (unweighted average of precision and recall of each class).

Decision Tree Max Depth 5

- Average accuracy: 0.956291390728
- Average recall: 0.922620674653
- Average precision: 0.946554566755

Decision Tree Max Depth 10

- Average accuracy: 0.949668874172
- Average recall: 0.936801663345
- Average precision: 0.937657827054

Decision Tree Max Features 5

- Average accuracy: 0.941059602649
- Average recall: 0.93292280777
- Average precision: 0.932324229589

Decision Tree Max Features 10

- Average accuracy: 0.955629139073
- Average recall: 0.940426030849
- Average precision: 0.946559870267

Linear SVC

- Average accuracy: 0.922516556291
- Average recall: 0.884663565109
- Average precision: 0.914970242306

Overall, the accuracy that we got for new activity is consistent with what we got for training the classifier with two activities only.

EXTRA CREDIT
Proper Methodology for Evaluating Model Performance

- We split the training and testing test set with a ratio of 4:1 and then trained 5 models on training data with 10-fold cross validation to find the model with best performance.
- With 80% of data, Decision Tree with max_depth of 5 yields the highest average accuracy of 0.95, average recall of 0.97 and average precision of 0.96.
- We then fit the model again with the all training data set and test it on the held-out testing data. Overall, we got:

    - **Accuracy** on held-out data: 0.95
    - **Precision** on held-out data: 0.96
    - **Recall** on held-out data: 0.96

- The performance is consistent with the statistic we got from while training model which confirm that our model does not overfit the data.

EXTRA CREDIT
Parameter Learning

- In this section, we find the best parameter for decision tree and support vector machine.
-

Decision Tree

Parameter search space

```
tuned_parameters = [{
    'criterion': ["gini", "entropy"],
    'max_depth': range(3, 22, 3),
    'max_features': range(3, 22, 3),
}]
```

Best parameter

{'max_features': 15, 'criterion': 'entropy', 'max_depth': 6}

Result

- Accuracy on held-out data: 0.95
- Recall on held-out data: 0.95
- Precision on held-out data: 0.97

Support Vector Machine

Parameter search space

```
tuned_parameters = [{
    'kernel': ['rbf'],
    'gamma': [1e-2, 1e-3, 1e-4],
    'C': [0.1, 1, 10, 100, 1000]
}, {
    'kernel': ['linear'],
    'C': [0.1, 1, 10]
}]
```

Best parameter

{'kernel': 'rbf', 'C': 1000, 'gamma': 0.0001}

Result

- Accuracy on held-out data: 0.96
- Recall on held-out data: 0.98
- Precision on held-out data: 0.95

Both Decision Tree and SVM with tuned parameters performed better than all of the classifier we experiment in the previous section. Overall, SVM with rbf kernel, c = 1000 and gamma = 0.0001 yields the best result with accuracy of 0.96, recall of 0.98 and precision of 0.95.

Member Roles & Contributions

- **Part 1**: Thai and Gahyun worked on computing the accuracy, precision, and recall metrics from the confusion matrix. Gahyun worked on calculating and printing the average accuracy, precision, and recall across all ten folds. Thai then handled the NaN cases in the calculation of the averages.

- **Part 2**: Thai extracted features including six statistical features from raw data and magnitude as well entropy from the histogram and created three graphs to visualize the relationship between features. Gahyun analyzed the visualizations and described which features worked best at discriminating activities.

- **Part 3**: Thai trained the 5 models with different parameters and report the accuracy, precision and recall. He also created the visualization for the tree. Colin analyzed the results and drew conclusions for the write-up.

- **Part 4**: Gahyun and Colin worked on the UI. Gahyun created a spinner for the user to select the activity and created the options (Sitting, Walking, Running, Jumping). Colin created a button for the user to start and stop recording the user's selected activity. Thai worked on sending labelled data to the server. Gahyun and Thai tested that the correct labels were being sent and received on the server. Gahyun and Colin collected data for each of the four selected activities. Thai used this data to train the classifier and send prediction back to the Android client.

- **Extra Credit:** Thai implemented and reported result for both Proper Methodology for Evaluating Model Performance and Parameter Learning