Exercise 9: Support Vector Machines

CPSC 381/581: Machine Learning

Yale University

Instructor: Alex Wong

Prerequisites:

1. Enable Google Colaboratory as an app on your Google Drive account
2. Create a new Google Colab notebook, this will also create a "Colab Notebooks" directory under "MyDrive" i.e.

   ```
   /content/drive/MyDrive/Colab Notebooks
   ```

3. Create the following directory structure in your Google Drive

   ```
   /content/drive/MyDrive/Colab Notebooks/CPSC 381-581: Machine
   Learning/Exercises
   ```

4. Move the 09_exercise_support_vector_machines.ipynb into

   ```
   /content/drive/MyDrive/Colab Notebooks/CPSC 381-581: Machine
   Learning/Exercises
   ```

so that its absolute path is

```
/content/drive/MyDrive/Colab Notebooks/CPSC 381-581: Machine
Learning/Exercises/09_exercise_support_vector_machines.ipynb
```

In this exercise, we will optimize a perceptron, logistic regression, and support vector machine on 4 datasets. We can compare classification accuracy across all the datasets to see which method is the best.

Submission:

1. Implement all TODOs in the code blocks below.
2. Report your validation scores for each method averaged over 10 trials.

   ```
   ***** Mean accuracy across 10 trials *****
   Perceptron: 0.7937024853801169
   Logistic Regression: 0.9523538011695907
   Support Vector Machine: 0.9668019005847952
   ```

3. List any collaborators.

   ```
   None
   ```

Import packages

In [ ]:
```
import numpy as np
import sklearn.datasets as skdata
import sklearn.metrics as skmetrics
```

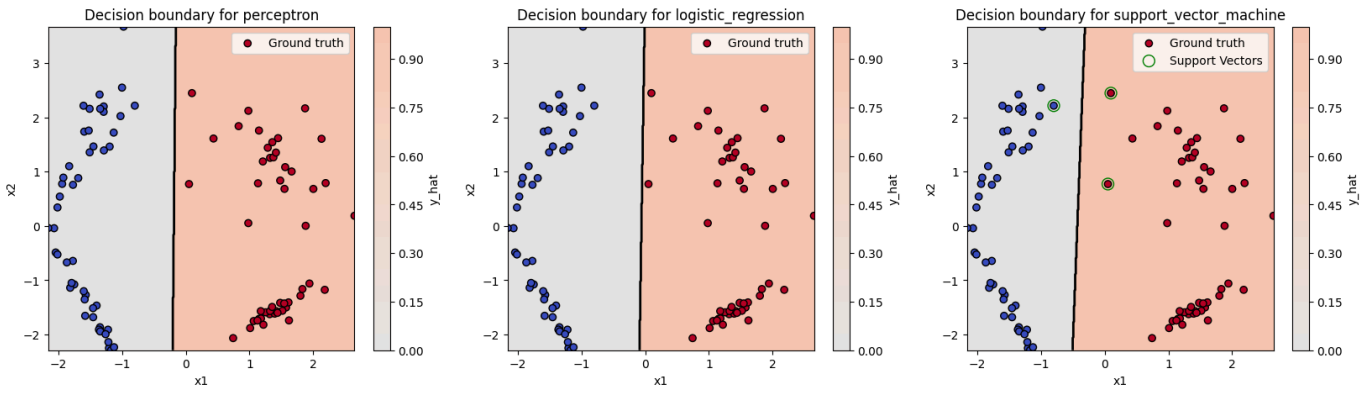
```
from sklearn.svm import SVC
from sklearn.linear_model import Perceptron
from sklearn.linear_model import LogisticRegression
from matplotlib import pyplot as plt
import warnings

warnings.filterwarnings(action='ignore')
np.random.seed = 1
```

Visualize the decision boundary and support vectors of different classifiers

In [ ]:

```
# Generate synthetic data
X, y = skdata.make_classification(
    n_features=2,
    n_classes=2,
    n_redundant=0,
    n_clusters_per_class=2,
    n_samples=100,
    class_sep=1.5,
    random_state=1)

methods = [
    'perceptron',
    'logistic_regression',
    'support_vector_machine'
]

# TODO: Create figure with figsize=(20, 5)
fig = plt.figure(figsize=(20, 5))

# TODO: Enumerate through methods with index

for i, method in enumerate(methods):
    # Instantiate model
    if method == 'perceptron':

        # TODO: Instantiate perceptron model with tolerance of 1e-1 and alpha of 0
        model = Perceptron(tol=1e-1, alpha=0)

    elif method == 'logistic_regression':

        # TODO: Instantiate logistic regression model with tolerance of 1e-1
        model = LogisticRegression(tol=1e-1)

    elif method == 'support_vector_machine':

        # TODO: Instantiate SVC (Support Vector Machine Classifier) with tolerance of
        model = SVC(tol=1e-1, C=1e10, kernel='linear')

    else:
        raise ValueError('Unsupported method: {}'.format(method))

    # TODO: Train the model
    model.fit(X, y)


    # TODO: Get x1_min and x1_max (0-th dimension), and x2_min and x2_max (1-st dimen
    x1_min, x1_max = min(X[:, 0]), max(X[:, 0])
    x2_min, x2_max = min(X[:, 1]), max(X[:, 1])

    # TODO: Create 2 linspaces: one from x1_min to x1_max and the other from x1_min t
    x1_linspace = np.linspace(x1_min, x1_max, 500)
    x2_linspace = np.linspace(x2_min, x2_max, 500)

    # TODO: Create meshgrid for x1 and x2 using linspaces
```

```
    x1, x2 = np.meshgrid(x1_linspace, x2_linspace)

    # TODO: Predict values for every point in meshgrid
    all_Xs = np.c_[x1.ravel(), x2.ravel()]
    y_hat = model.predict(all_Xs)

    # TODO: Reshape y_hat to x1 or x2's shape
    y_hat = y_hat.reshape(x1.shape)

    # TODO: Instantiate axis for subplot of a 1 x 3 figure
    ax = fig.add_subplot(1, 3, i + 1)

    # TODO: Plot Contour for predictions with levels=20, cmap='coolwarm', alpha=0.8,
    contour = ax.contourf(x1, x2, y_hat, levels=20, cmap='coolwarm', alpha=0.8, vmin=

    # TODO: Create colorbar for contour on axis and set its label to 'y_hat'
    cbar = fig.colorbar(contour, ax=ax, label='y_hat')

    # TODO: Plot decision boundary using levels=[0], colors='black', linewidths=2
    decision_boundary = ax.contour(x1, x2, y_hat, levels=[0], colors='black', linewid

    # TODO: Create scatter plot for X and set its color to y with edgecolor='black',
    ax.scatter(X[:, 0], X[:, 1], c=y, edgecolor='black', cmap='coolwarm', label='Grou

    # TODO: If support vector machine
    # Create scatter plot of support vectors with s=100, facecolors='none', edgecolor
    if method == 'support_vector_machine':
        ax.scatter(model.support_vectors_[:, 0], model.support_vectors_[:, 1], s=100,

    # TODO: Set title to ''Decision boundary for {}'

    ax.set_title('Decision boundary for {}'.format(method))

    # TODO: Set xlabel to 'x1'

    ax.set_xlabel('x1')

    # TODO: Set ylabel to 'x2'
    ax.set_ylabel('x2')

    # TODO: Set legend with loc='upper right'
    ax.legend(loc='upper right')

plt.show()
```



Load datasets

In [ ]:

```
# Load datasets
datasets = [
    skdata.load_iris(),
    skdata.load_breast_cancer(),
    skdata.load_digits(),
    skdata.load_wine()
```

```
]

dataset_names = [
    'Iris',
    'Breast cancer',
    'Digits',
    'Wine'
]
```

Compare Perceptron, Logistic Regression, and Support Vector Machines across all datasets

In [ ]:

```
# Define lists to hold validation scores across trials
scores_val = {
    'perceptron' : [],
    'logistic_regression': [],
    'support_vector_machine' : []
}

# Perform 10 trials of experiments
n_trial = 10

for n in range(n_trial):

    print('***** TRIAL {} *****\n'.format(n))

    # Zip up all dataset options
    dataset_options = zip(
        datasets,
        dataset_names)

    for dataset, dataset_name in dataset_options:

        '''
        Create the training and validation splits
        '''
        X = dataset.data
        y = dataset.target

        print('Preprocessing the {} dataset ({} samples, {} feature dimensions)'.form

        # Shuffle the dataset based on sample indices
        shuffled_indices = np.random.permutation(X.shape[0])

        # Choose the first 80% as training set and the next 20% as validation
        train_split_idx = int(0.80 * X.shape[0])

        train_indices = shuffled_indices[0:train_split_idx]
        val_indices = shuffled_indices[train_split_idx:]

        # Select the examples from X and y to construct our training and validation s
        X_train, y_train = X[train_indices, :], y[train_indices]
        X_val, y_val = X[val_indices, :], y[val_indices]

        for method in ['perceptron', 'logistic_regression', 'support_vector_machine']

            print('***** Experiments on the {} dataset using {} model *****'.format(
                dataset_name,
                method))

            # Instantiate model
            if method == 'perceptron':

                # TODO: Instantiate perceptron model with tolerance of 1e-1 and alpha
                model = Perceptron(tol=1e-1, alpha=0)
```

```
            elif method == 'logistic_regression':

                # TODO: Instantiate logistic regression model with tolerance of 1e-1
                model = LogisticRegression(tol=1e-1)

            elif method == 'support_vector_machine':

                # TODO: Instantiate SVC (Support Vector Machine Classifier) with tole
                model = SVC(tol=1e-1, C=1, kernel='linear')

            else:
                raise ValueError('Unsupported method: {}'.format(method))


            # TODO: Train the model
            model.fit(X_train, y_train)

            # TODO: Score model using mean accuracy on training set
            predictions_train = skmetrics.accuracy_score(y_train, model.predict(X_tra
            score_train = skmetrics.accuracy_score(y_train, model.predict(X_train))
            print('Training set mean accuracy: {:.4f}'.format(score_train))

            # TODO: Score model using mean accuracy validation set
            predictions_val = skmetrics.accuracy_score(y_val, model.predict(X_val))
            score_val = skmetrics.accuracy_score(y_val, model.predict(X_val))
            print('Validation set mean accuracy: {:.4f}'.format(score_val))

            # TODO: Append score to validation scores for the given method
            scores_val[method].append(score_val)

        print('')

# TODO: Compute mean over trials for each method
mean_scores_val_perceptron = np.mean(scores_val['perceptron'])
mean_scores_val_logistic = np.mean(scores_val['logistic_regression'])
mean_scores_val_svm = np.mean(scores_val['support_vector_machine'])

print('***** Mean accuracy across {} trials *****'.format(n_trial))

print('Perceptron: {}'.format(mean_scores_val_perceptron))
print('Logistic Regression: {}'.format(mean_scores_val_logistic))
print('Support Vector Machine: {}'.format(mean_scores_val_svm))
```

```
***** TRIAL 0 *****

Preprocessing the Iris dataset (150 samples, 4 feature dimensions)
***** Experiments on the Iris dataset using perceptron model *****
Training set mean accuracy: 0.9500
Validation set mean accuracy: 0.9333
***** Experiments on the Iris dataset using logistic_regression model *****
Training set mean accuracy: 0.9667
Validation set mean accuracy: 1.0000
***** Experiments on the Iris dataset using support_vector_machine model *****
Training set mean accuracy: 0.9833
Validation set mean accuracy: 1.0000

Preprocessing the Breast cancer dataset (569 samples, 30 feature dimensions)
***** Experiments on the Breast cancer dataset using perceptron model *****
Training set mean accuracy: 0.8989
Validation set mean accuracy: 0.9561
***** Experiments on the Breast cancer dataset using logistic_regression model *****
Training set mean accuracy: 0.9231
Validation set mean accuracy: 0.9737
***** Experiments on the Breast cancer dataset using support_vector_machine model ****
*
Training set mean accuracy: 0.9604
Validation set mean accuracy: 0.9825

Preprocessing the Digits dataset (1797 samples, 64 feature dimensions)
***** Experiments on the Digits dataset using perceptron model *****
Training set mean accuracy: 0.9784
Validation set mean accuracy: 0.9472
***** Experiments on the Digits dataset using logistic_regression model *****
Training set mean accuracy: 0.9603
Validation set mean accuracy: 0.9528
***** Experiments on the Digits dataset using support_vector_machine model *****
Training set mean accuracy: 1.0000
Validation set mean accuracy: 0.9694

Preprocessing the Wine dataset (178 samples, 13 feature dimensions)
***** Experiments on the Wine dataset using perceptron model *****
Training set mean accuracy: 0.6408
Validation set mean accuracy: 0.7500
***** Experiments on the Wine dataset using logistic_regression model *****
Training set mean accuracy: 0.9225
Validation set mean accuracy: 1.0000
***** Experiments on the Wine dataset using support_vector_machine model *****
Training set mean accuracy: 0.9930
Validation set mean accuracy: 0.9722

***** TRIAL 1 *****

Preprocessing the Iris dataset (150 samples, 4 feature dimensions)
***** Experiments on the Iris dataset using perceptron model *****
Training set mean accuracy: 0.9417
Validation set mean accuracy: 0.9333
***** Experiments on the Iris dataset using logistic_regression model *****
Training set mean accuracy: 0.9583
Validation set mean accuracy: 0.9000
***** Experiments on the Iris dataset using support_vector_machine model *****
Training set mean accuracy: 0.9917
Validation set mean accuracy: 0.9333

Preprocessing the Breast cancer dataset (569 samples, 30 feature dimensions)
***** Experiments on the Breast cancer dataset using perceptron model *****
Training set mean accuracy: 0.8945
Validation set mean accuracy: 0.9123
***** Experiments on the Breast cancer dataset using logistic_regression model *****
Training set mean accuracy: 0.9275
```

```
Validation set mean accuracy: 0.9561
***** Experiments on the Breast cancer dataset using support_vector_machine model ****
*
Training set mean accuracy: 0.9670
Validation set mean accuracy: 0.9561

Preprocessing the Digits dataset (1797 samples, 64 feature dimensions)
***** Experiments on the Digits dataset using perceptron model *****
Training set mean accuracy: 0.9673
Validation set mean accuracy: 0.9694
***** Experiments on the Digits dataset using logistic_regression model *****
Training set mean accuracy: 0.9569
Validation set mean accuracy: 0.9694
***** Experiments on the Digits dataset using support_vector_machine model *****
Training set mean accuracy: 1.0000
Validation set mean accuracy: 1.0000

Preprocessing the Wine dataset (178 samples, 13 feature dimensions)
***** Experiments on the Wine dataset using perceptron model *****
Training set mean accuracy: 0.6479
Validation set mean accuracy: 0.6667
***** Experiments on the Wine dataset using logistic_regression model *****
Training set mean accuracy: 0.9296
Validation set mean accuracy: 0.9444
***** Experiments on the Wine dataset using support_vector_machine model *****
Training set mean accuracy: 1.0000
Validation set mean accuracy: 0.9444

***** TRIAL 2 *****

Preprocessing the Iris dataset (150 samples, 4 feature dimensions)
***** Experiments on the Iris dataset using perceptron model *****
Training set mean accuracy: 0.6583
Validation set mean accuracy: 0.7000
***** Experiments on the Iris dataset using logistic_regression model *****
Training set mean accuracy: 0.9667
Validation set mean accuracy: 1.0000
***** Experiments on the Iris dataset using support_vector_machine model *****
Training set mean accuracy: 0.9833
Validation set mean accuracy: 1.0000

Preprocessing the Breast cancer dataset (569 samples, 30 feature dimensions)
***** Experiments on the Breast cancer dataset using perceptron model *****
Training set mean accuracy: 0.8835
Validation set mean accuracy: 0.9386
***** Experiments on the Breast cancer dataset using logistic_regression model *****
Training set mean accuracy: 0.9275
Validation set mean accuracy: 0.9561
***** Experiments on the Breast cancer dataset using support_vector_machine model ****
*
Training set mean accuracy: 0.9582
Validation set mean accuracy: 0.9737

Preprocessing the Digits dataset (1797 samples, 64 feature dimensions)
***** Experiments on the Digits dataset using perceptron model *****
Training set mean accuracy: 0.9631
Validation set mean accuracy: 0.9528
***** Experiments on the Digits dataset using logistic_regression model *****
Training set mean accuracy: 0.9645
Validation set mean accuracy: 0.9500
***** Experiments on the Digits dataset using support_vector_machine model *****
Training set mean accuracy: 1.0000
Validation set mean accuracy: 0.9722

Preprocessing the Wine dataset (178 samples, 13 feature dimensions)
***** Experiments on the Wine dataset using perceptron model *****
```

```
Training set mean accuracy: 0.5493
Validation set mean accuracy: 0.6111
***** Experiments on the Wine dataset using logistic_regression model *****
Training set mean accuracy: 0.9648
Validation set mean accuracy: 0.9722
***** Experiments on the Wine dataset using support_vector_machine model *****
Training set mean accuracy: 0.9930
Validation set mean accuracy: 1.0000

***** TRIAL 3 *****

Preprocessing the Iris dataset (150 samples, 4 feature dimensions)
***** Experiments on the Iris dataset using perceptron model *****
Training set mean accuracy: 0.4750
Validation set mean accuracy: 0.6000
***** Experiments on the Iris dataset using logistic_regression model *****
Training set mean accuracy: 0.9667
Validation set mean accuracy: 1.0000
***** Experiments on the Iris dataset using support_vector_machine model *****
Training set mean accuracy: 0.9750
Validation set mean accuracy: 1.0000

Preprocessing the Breast cancer dataset (569 samples, 30 feature dimensions)
***** Experiments on the Breast cancer dataset using perceptron model *****
Training set mean accuracy: 0.9055
Validation set mean accuracy: 0.8947
***** Experiments on the Breast cancer dataset using logistic_regression model *****
Training set mean accuracy: 0.9341
Validation set mean accuracy: 0.9123
***** Experiments on the Breast cancer dataset using support_vector_machine model ****
*
Training set mean accuracy: 0.9758
Validation set mean accuracy: 0.9211

Preprocessing the Digits dataset (1797 samples, 64 feature dimensions)
***** Experiments on the Digits dataset using perceptron model *****
Training set mean accuracy: 0.9638
Validation set mean accuracy: 0.9250
***** Experiments on the Digits dataset using logistic_regression model *****
Training set mean accuracy: 0.9659
Validation set mean accuracy: 0.9444
***** Experiments on the Digits dataset using support_vector_machine model *****
Training set mean accuracy: 1.0000
Validation set mean accuracy: 0.9750

Preprocessing the Wine dataset (178 samples, 13 feature dimensions)
***** Experiments on the Wine dataset using perceptron model *****
Training set mean accuracy: 0.4014
Validation set mean accuracy: 0.3889
***** Experiments on the Wine dataset using logistic_regression model *****
Training set mean accuracy: 0.9789
Validation set mean accuracy: 0.9167
***** Experiments on the Wine dataset using support_vector_machine model *****
Training set mean accuracy: 1.0000
Validation set mean accuracy: 0.8889

***** TRIAL 4 *****

Preprocessing the Iris dataset (150 samples, 4 feature dimensions)
***** Experiments on the Iris dataset using perceptron model *****
Training set mean accuracy: 0.6833
Validation set mean accuracy: 0.6333
***** Experiments on the Iris dataset using logistic_regression model *****
Training set mean accuracy: 0.9667
Validation set mean accuracy: 0.9667
***** Experiments on the Iris dataset using support_vector_machine model *****
```

```
Training set mean accuracy: 0.9833
Validation set mean accuracy: 1.0000

Preprocessing the Breast cancer dataset (569 samples, 30 feature dimensions)
***** Experiments on the Breast cancer dataset using perceptron model *****
Training set mean accuracy: 0.7824
Validation set mean accuracy: 0.7895
***** Experiments on the Breast cancer dataset using logistic_regression model *****
Training set mean accuracy: 0.9253
Validation set mean accuracy: 0.9474
***** Experiments on the Breast cancer dataset using support_vector_machine model ****
*
Training set mean accuracy: 0.9714
Validation set mean accuracy: 0.9386

Preprocessing the Digits dataset (1797 samples, 64 feature dimensions)
***** Experiments on the Digits dataset using perceptron model *****
Training set mean accuracy: 0.9548
Validation set mean accuracy: 0.9500
***** Experiments on the Digits dataset using logistic_regression model *****
Training set mean accuracy: 0.9659
Validation set mean accuracy: 0.9667
***** Experiments on the Digits dataset using support_vector_machine model *****
Training set mean accuracy: 1.0000
Validation set mean accuracy: 0.9861

Preprocessing the Wine dataset (178 samples, 13 feature dimensions)
***** Experiments on the Wine dataset using perceptron model *****
Training set mean accuracy: 0.6338
Validation set mean accuracy: 0.6111
***** Experiments on the Wine dataset using logistic_regression model *****
Training set mean accuracy: 0.9648
Validation set mean accuracy: 0.9722
***** Experiments on the Wine dataset using support_vector_machine model *****
Training set mean accuracy: 1.0000
Validation set mean accuracy: 0.9722

***** TRIAL 5 *****

Preprocessing the Iris dataset (150 samples, 4 feature dimensions)
***** Experiments on the Iris dataset using perceptron model *****
Training set mean accuracy: 0.9500
Validation set mean accuracy: 1.0000
***** Experiments on the Iris dataset using logistic_regression model *****
Training set mean accuracy: 0.9667
Validation set mean accuracy: 1.0000
***** Experiments on the Iris dataset using support_vector_machine model *****
Training set mean accuracy: 0.9833
Validation set mean accuracy: 0.9667

Preprocessing the Breast cancer dataset (569 samples, 30 feature dimensions)
***** Experiments on the Breast cancer dataset using perceptron model *****
Training set mean accuracy: 0.9253
Validation set mean accuracy: 0.8860
***** Experiments on the Breast cancer dataset using logistic_regression model *****
Training set mean accuracy: 0.9297
Validation set mean accuracy: 0.9298
***** Experiments on the Breast cancer dataset using support_vector_machine model ****
*
Training set mean accuracy: 0.9604
Validation set mean accuracy: 0.9561

Preprocessing the Digits dataset (1797 samples, 64 feature dimensions)
***** Experiments on the Digits dataset using perceptron model *****
Training set mean accuracy: 0.9722
Validation set mean accuracy: 0.9528
```

```
***** Experiments on the Digits dataset using logistic_regression model *****
Training set mean accuracy: 0.9576
Validation set mean accuracy: 0.9611
***** Experiments on the Digits dataset using support_vector_machine model *****
Training set mean accuracy: 1.0000
Validation set mean accuracy: 0.9861

Preprocessing the Wine dataset (178 samples, 13 feature dimensions)
***** Experiments on the Wine dataset using perceptron model *****
Training set mean accuracy: 0.6549
Validation set mean accuracy: 0.6111
***** Experiments on the Wine dataset using logistic_regression model *****
Training set mean accuracy: 0.9577
Validation set mean accuracy: 0.9167
***** Experiments on the Wine dataset using support_vector_machine model *****
Training set mean accuracy: 1.0000
Validation set mean accuracy: 0.9444

***** TRIAL 6 *****

Preprocessing the Iris dataset (150 samples, 4 feature dimensions)
***** Experiments on the Iris dataset using perceptron model *****
Training set mean accuracy: 0.7500
Validation set mean accuracy: 0.8000
***** Experiments on the Iris dataset using logistic_regression model *****
Training set mean accuracy: 0.9750
Validation set mean accuracy: 0.9667
***** Experiments on the Iris dataset using support_vector_machine model *****
Training set mean accuracy: 0.9833
Validation set mean accuracy: 0.9667

Preprocessing the Breast cancer dataset (569 samples, 30 feature dimensions)
***** Experiments on the Breast cancer dataset using perceptron model *****
Training set mean accuracy: 0.8374
Validation set mean accuracy: 0.7193
***** Experiments on the Breast cancer dataset using logistic_regression model *****
Training set mean accuracy: 0.9319
Validation set mean accuracy: 0.9386
***** Experiments on the Breast cancer dataset using support_vector_machine model ****
*
Training set mean accuracy: 0.9736
Validation set mean accuracy: 0.9298

Preprocessing the Digits dataset (1797 samples, 64 feature dimensions)
***** Experiments on the Digits dataset using perceptron model *****
Training set mean accuracy: 0.9687
Validation set mean accuracy: 0.9444
***** Experiments on the Digits dataset using logistic_regression model *****
Training set mean accuracy: 0.9652
Validation set mean accuracy: 0.9583
***** Experiments on the Digits dataset using support_vector_machine model *****
Training set mean accuracy: 1.0000
Validation set mean accuracy: 0.9722

Preprocessing the Wine dataset (178 samples, 13 feature dimensions)
***** Experiments on the Wine dataset using perceptron model *****
Training set mean accuracy: 0.5704
Validation set mean accuracy: 0.6944
***** Experiments on the Wine dataset using logistic_regression model *****
Training set mean accuracy: 0.9718
Validation set mean accuracy: 0.8611
***** Experiments on the Wine dataset using support_vector_machine model *****
Training set mean accuracy: 1.0000
Validation set mean accuracy: 0.8889

***** TRIAL 7 *****
```

```
Preprocessing the Iris dataset (150 samples, 4 feature dimensions)
***** Experiments on the Iris dataset using perceptron model *****
Training set mean accuracy: 0.6500
Validation set mean accuracy: 0.4000
***** Experiments on the Iris dataset using logistic_regression model *****
Training set mean accuracy: 0.9750
Validation set mean accuracy: 0.9667
***** Experiments on the Iris dataset using support_vector_machine model *****
Training set mean accuracy: 0.9750
Validation set mean accuracy: 1.0000

Preprocessing the Breast cancer dataset (569 samples, 30 feature dimensions)
***** Experiments on the Breast cancer dataset using perceptron model *****
Training set mean accuracy: 0.9143
Validation set mean accuracy: 0.9298
***** Experiments on the Breast cancer dataset using logistic_regression model *****
Training set mean accuracy: 0.9363
Validation set mean accuracy: 0.9298
***** Experiments on the Breast cancer dataset using support_vector_machine model ****
*
Training set mean accuracy: 0.9648
Validation set mean accuracy: 0.9737

Preprocessing the Digits dataset (1797 samples, 64 feature dimensions)
***** Experiments on the Digits dataset using perceptron model *****
Training set mean accuracy: 0.9645
Validation set mean accuracy: 0.9694
***** Experiments on the Digits dataset using logistic_regression model *****
Training set mean accuracy: 0.9666
Validation set mean accuracy: 0.9750
***** Experiments on the Digits dataset using support_vector_machine model *****
Training set mean accuracy: 1.0000
Validation set mean accuracy: 0.9833

Preprocessing the Wine dataset (178 samples, 13 feature dimensions)
***** Experiments on the Wine dataset using perceptron model *****
Training set mean accuracy: 0.5704
Validation set mean accuracy: 0.7222
***** Experiments on the Wine dataset using logistic_regression model *****
Training set mean accuracy: 0.9225
Validation set mean accuracy: 0.9444
***** Experiments on the Wine dataset using support_vector_machine model *****
Training set mean accuracy: 0.9930
Validation set mean accuracy: 1.0000

***** TRIAL 8 *****

Preprocessing the Iris dataset (150 samples, 4 feature dimensions)
***** Experiments on the Iris dataset using perceptron model *****
Training set mean accuracy: 0.4000
Validation set mean accuracy: 0.4667
***** Experiments on the Iris dataset using logistic_regression model *****
Training set mean accuracy: 0.9833
Validation set mean accuracy: 0.9000
***** Experiments on the Iris dataset using support_vector_machine model *****
Training set mean accuracy: 0.9833
Validation set mean accuracy: 1.0000

Preprocessing the Breast cancer dataset (569 samples, 30 feature dimensions)
***** Experiments on the Breast cancer dataset using perceptron model *****
Training set mean accuracy: 0.8747
Validation set mean accuracy: 0.9035
***** Experiments on the Breast cancer dataset using logistic_regression model *****
Training set mean accuracy: 0.9319
Validation set mean accuracy: 0.9386
```

```
***** Experiments on the Breast cancer dataset using support_vector_machine model ****
*
Training set mean accuracy: 0.9692
Validation set mean accuracy: 0.9386

Preprocessing the Digits dataset (1797 samples, 64 feature dimensions)
***** Experiments on the Digits dataset using perceptron model *****
Training set mean accuracy: 0.9798
Validation set mean accuracy: 0.9556
***** Experiments on the Digits dataset using logistic_regression model *****
Training set mean accuracy: 0.9673
Validation set mean accuracy: 0.9417
***** Experiments on the Digits dataset using support_vector_machine model *****
Training set mean accuracy: 1.0000
Validation set mean accuracy: 0.9639

Preprocessing the Wine dataset (178 samples, 13 feature dimensions)
***** Experiments on the Wine dataset using perceptron model *****
Training set mean accuracy: 0.6338
Validation set mean accuracy: 0.6111
***** Experiments on the Wine dataset using logistic_regression model *****
Training set mean accuracy: 0.9507
Validation set mean accuracy: 0.9444
***** Experiments on the Wine dataset using support_vector_machine model *****
Training set mean accuracy: 1.0000
Validation set mean accuracy: 0.8889

***** TRIAL 9 *****

Preprocessing the Iris dataset (150 samples, 4 feature dimensions)
***** Experiments on the Iris dataset using perceptron model *****
Training set mean accuracy: 0.6583
Validation set mean accuracy: 0.7000
***** Experiments on the Iris dataset using logistic_regression model *****
Training set mean accuracy: 0.9750
Validation set mean accuracy: 0.9667
***** Experiments on the Iris dataset using support_vector_machine model *****
Training set mean accuracy: 0.9917
Validation set mean accuracy: 0.9667

Preprocessing the Breast cancer dataset (569 samples, 30 feature dimensions)
***** Experiments on the Breast cancer dataset using perceptron model *****
Training set mean accuracy: 0.9187
Validation set mean accuracy: 0.9211
***** Experiments on the Breast cancer dataset using logistic_regression model *****
Training set mean accuracy: 0.9275
Validation set mean accuracy: 0.9561
***** Experiments on the Breast cancer dataset using support_vector_machine model ****
*
Training set mean accuracy: 0.9604
Validation set mean accuracy: 0.9825

Preprocessing the Digits dataset (1797 samples, 64 feature dimensions)
***** Experiments on the Digits dataset using perceptron model *****
Training set mean accuracy: 0.9562
Validation set mean accuracy: 0.9417
***** Experiments on the Digits dataset using logistic_regression model *****
Training set mean accuracy: 0.9555
Validation set mean accuracy: 0.9528
***** Experiments on the Digits dataset using support_vector_machine model *****
Training set mean accuracy: 1.0000
Validation set mean accuracy: 0.9778

Preprocessing the Wine dataset (178 samples, 13 feature dimensions)
***** Experiments on the Wine dataset using perceptron model *****
Training set mean accuracy: 0.6831
```

```
Validation set mean accuracy: 0.5556
***** Experiments on the Wine dataset using logistic_regression model *****
Training set mean accuracy: 0.9648
Validation set mean accuracy: 0.9444
***** Experiments on the Wine dataset using support_vector_machine model *****
Training set mean accuracy: 0.9930
Validation set mean accuracy: 1.0000

***** Mean accuracy across 10 trials *****
Perceptron: 0.7937024853801169
Logistic Regression: 0.9523538011695907
Support Vector Machine: 0.9668019005847952
```