Max Kramer

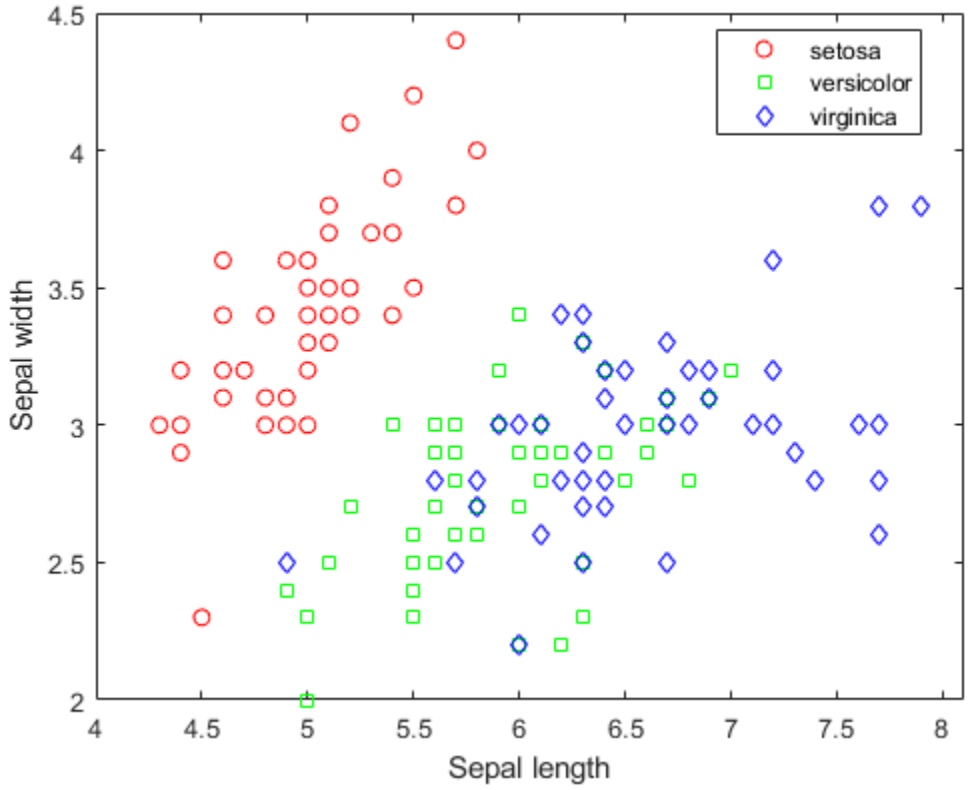
I affirm that I have adhered to the honor code on this assignment

Hello again, scientist! I'll do all my writing in italics, and problems for you will be in ***bold.*** *Comment your code, and* explain your ideas in plaintext. *As a general rule, I expect you to do at least as much writing as I do. Code should be part of your solution, but I expect variables to be clear and explanation to involve complete sentences. Cite your sources; if you work with someone in the class on a problem, that's an extremely important source. Don't work alone.*

Problem F.07: Separate the flowers.

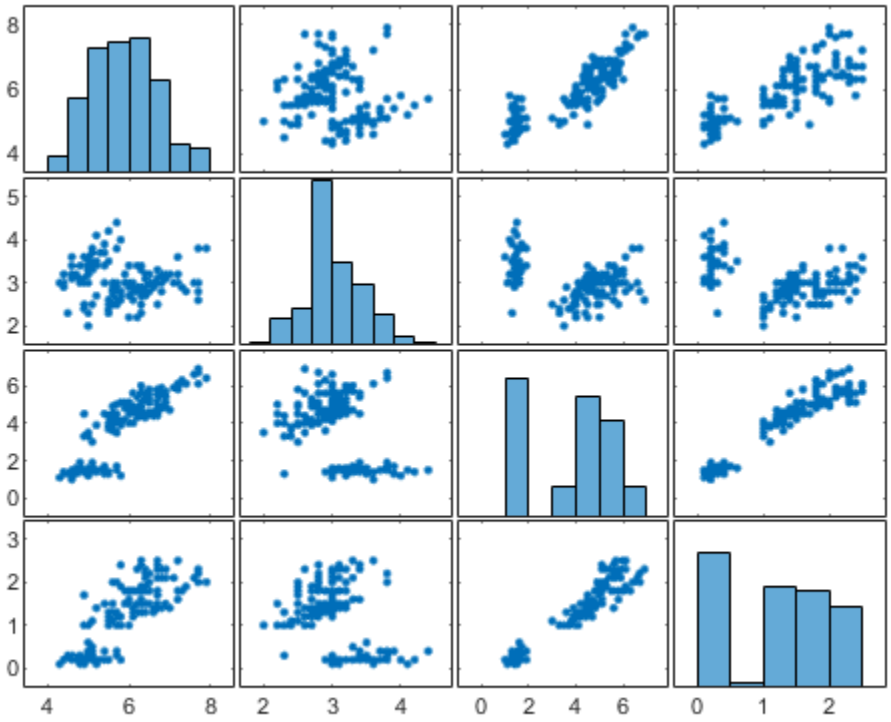
This problem assumes that you have completed most of F.04 and F.06. *This is the oldest and most famous data set in machine learning. https://onlinelibrary.wiley.com/doi/epdf/10.1111/j.1469-1809.1936.tb02137.x This article remains incredibly readable. For those of you who've taken Calc III: note the brilliant use of Lagrange multipliers on p.181.* ***Read about the data set: https://en.wikipedia.org/wiki/Iris_flower_data_set. (Just read the introduction, and look at the pictures of the flowers.)***

```
load iris;
figure;
gscatter(meas(:,1), meas(:,2), species,'rgb','osd');
xlabel('Sepal length');
ylabel('Sepal width');
```



The following command is also useful -- it lets you see every axis against every other. Notice that the picture plotted above is exactly the one in the first column and second row of the one below.

```
figure;
plotmatrix(meas)
```



Is flower #47 Iris setosa, Iris virginica, or Iris versicolor? What are the sepal and petal lengths of flower #47?

```
species_47 = species(47)
sepal_len_47 = meas(47,1)
sepal_wid_47 = meas(47,2)
```

```
species_47 =

  1×1 cell array

    {'setosa'}


sepal_len_47 =

    5.1000


sepal_wid_47 =
```

```
    3.8000
```

We can find the species of flower 47 by indexing into the cell array at index 47. The returned value is setosa, so the flower is Iris setose. We know from the gscatter that meas(:,1) is sepal length and meas(:,2) is sepal width, so indexing into those vectors at element 47 gives us the sepal length and width of flower 47.

Fisher found that it was possible to linearly separate the species Iris setosa (the red circles) from the other two species. He invented what's now called the "Support Vector Machine" (SVM), which remains the premier way to pull classes of data apart. What we're about to do isn't exactly an SVM, but the idea is the same: we're going to find a linear function that cuts the data into two distinct chunks. (The difference, for those of you who care about machine learning: PCA is "unsupervised," meaning that it doesn't actually know that the red circles are supposed to be distinct from the blue and green ones. SVM is "supervised," meaning that it starts out by knowing that the red circles are supposed to be different from the blue and green ones and then actively tries to classify the whole space as "red space" or "non-red space" using the data that it has.)

I'm about to perform principal component analysis (PCA) on meas. Pay attention. ***Don't call pca() for any MATLAB problems this semester.*** *That command is useful but haunted. This is my course; we use SVDs to do PCA. (That's what MATLAB does under the hood anyway.)*

```
m = mean(meas); % m is for mean
M = meas-m; % this mean-centers the data
[U,S,V]=svd(M,'econ'); % this is the PCA, it's just the SVD of mean-
centered data
diag(S)' % these are the singular values of M
```

```
ans =

   25.1000    6.0131    3.4137    1.8845
```

Show me that over 90% of the variance is explained by the first principal component. *That's a good thing, since it means that we can explain most of the variation in the data using a single axis.*

```
svals = diag(S);
vpa(sum(svals(1).^2)/sum(svals.^2),6)
```

```
ans =

0.924619
```

Svals is a vector containing the singular values of M. the vpa command calculates the proportion of variance explained by the first principal component. The proportion of variance explained by the first principal component in this case is 92.4619%.

*Let's take a look at that first principal component. Since U*S = A*V, the first column of U*S is the first column of A*V, which is A*V(:,1). This means that V(:,1) is the axis which explains the greatest possible amount of variation in the data. It also means that the coefficients of V(:,1) tell you how to construct the rotated data as a linear combination of the original features. These coefficients are called loadings.*

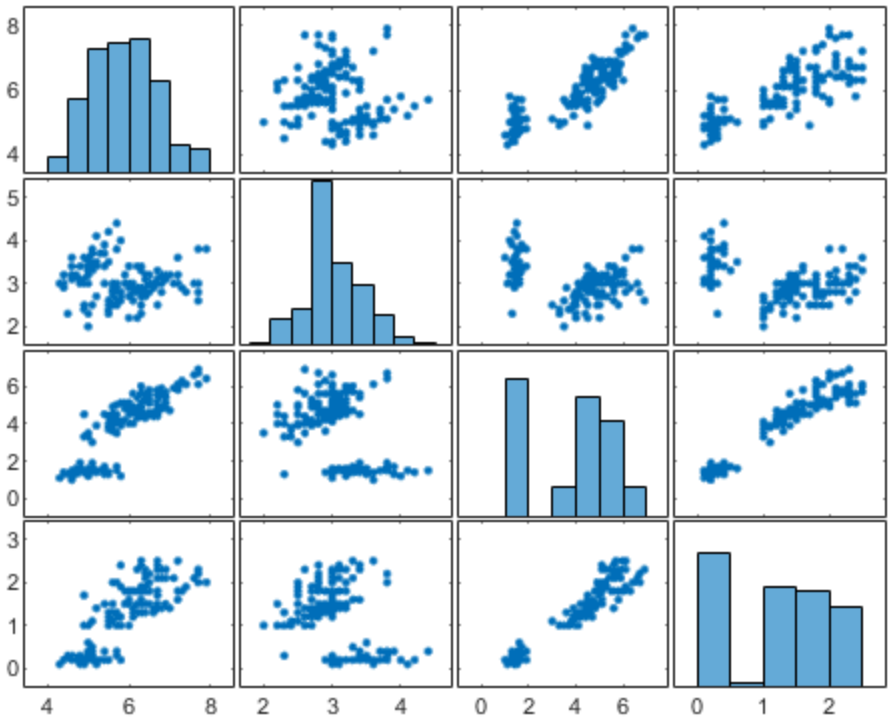
```
V(:,1)
```

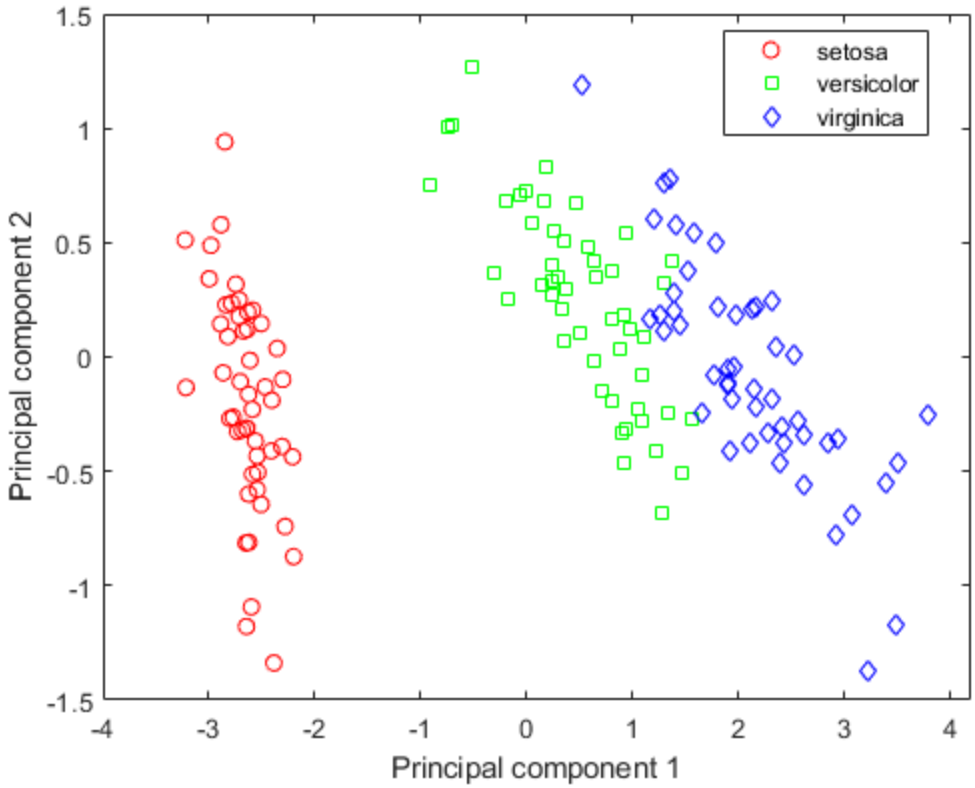
```
ans =

    0.3614
   -0.0845
    0.8567
    0.3583
```

What this says is that the first principal component is strongly positively correlated with petal length, somewhat positively correlated with sepal length and petal width, and pretty much ignores sepal width.

Okay, now let's rotate the data.

```
P = M*V;
figure;
gscatter(P(:,1), P(:,2), species,'rgb','osd');
xlabel('Principal component 1');
ylabel('Principal component 2');
```





Well dang look at that! We've separated the flowers. The coefficients of the rotated data are called the component scores. Since this data is effectively one-dimensional, we'll only keep the first dimension and look at the first score, which we'll call the flower's F-score. (F is for flower.) This gives a criterion for telling the flowers apart: if the F-score of a flower is less than -2, it must be Iris setosa. If its F-score is more than -1, it must not be Iris setosa. ***Find the F-score of flower #47.*** *(Since flower #47 is Iris setosa, its F-score should be less than -2.)*

Now let's do some classification. Here's the measurements of two unknown flowers f1 and f2.

```
f1 = [6 3 5 2];
f2 = [5 3 2 .5];
```

One of these flowers is Iris setosa; the other is not. ***Which is which? Compute their F-scores.*** *If you're getting thoroughly weird and broken numbers, remember that PCA only applies to mean-centered data.*

```
M2 = f1 - m;
P2 = M2 * V;
F_f1 = P2(1)

M3 = f2 - m;
P3 = M3 * V;
F_f2 = P3(1)
```

```
F_f1 =

    1.4123
```

```
F_f2 =

   -2.0565
```

Each sample is mean centered by subtracting the population mean m from the sample. The SVD is then calculated for each sample and then the sample is rotated. The resulting F scores indicate that f2 (-2.0565) is setosa and f1 (1.4123) is not.

Let's do a little genetic engineering. ***Come up with reasonable-seeming dimensions for a flower with an F-score of close to -1.5.*** *(Maybe try a hybrid?)*

```
f3 = [5.5 3.2 2.5 0.6];
M4 = f3 - m;
P4 = M4 * V;
F_f4 = P4(1)
```

```
F_f4 =

   -1.4286
```

To produce a flower with an F score of approximately -1.5, we need a hybrid between a setosa and versicolor. By examining the highest F scored setosa and the lowest F scored versicolor and averaging their measurements, we found an example that produced a first component score of -1.4286.

Published with MATLAB® R2019b