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.09: Factor the personalities.

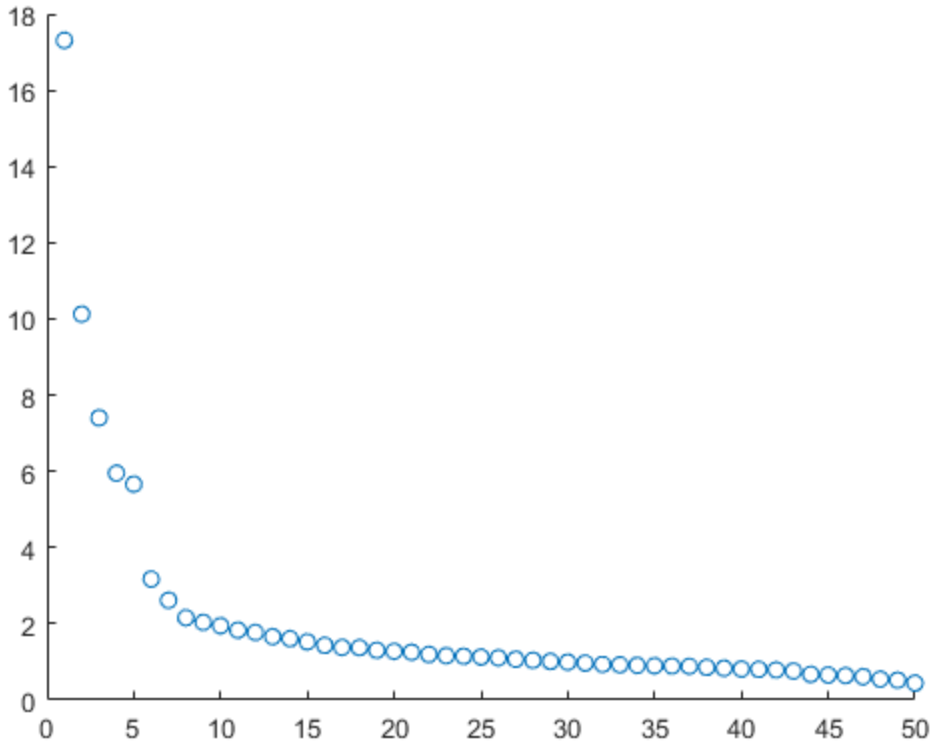
This problem assumes that you have mostly completed F.07 and F.08. *Let's look at some big data.* ***Read this: https://openpsychometrics.org/tests/IPIP-BFFM/ Take it yourself if you want, it's pretty quick!***

```
B = csvread('big5data.csv'); B=B-mean(B); [U,S,V]=svd(B,'econ');
size(B)
```

```
ans =

       19719          50
```

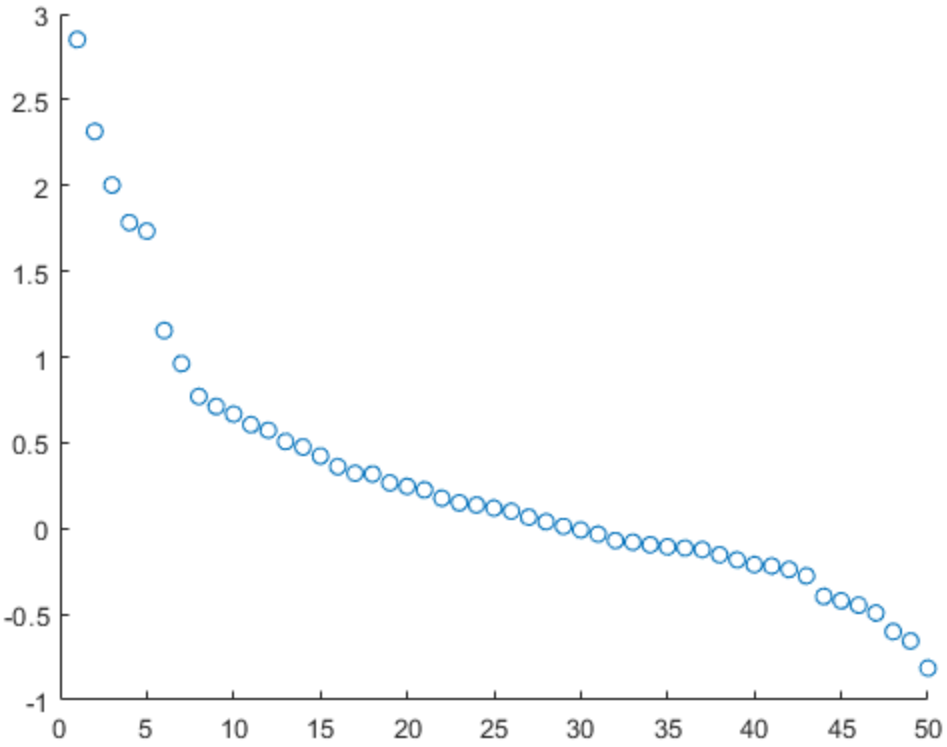
A few years ago, the Open-Source Psychometrics Project compiled 20,000 responses to the Big Five Personality Test into a single csv file. That's B. What we've just done is a PCA on B. If this is a good model, we expect that the data should be meaningfully five-dimensional. Let's see.

```
var = diag(S).^2; var=var/sum(var)*100;

scatter(1:50,var)
```



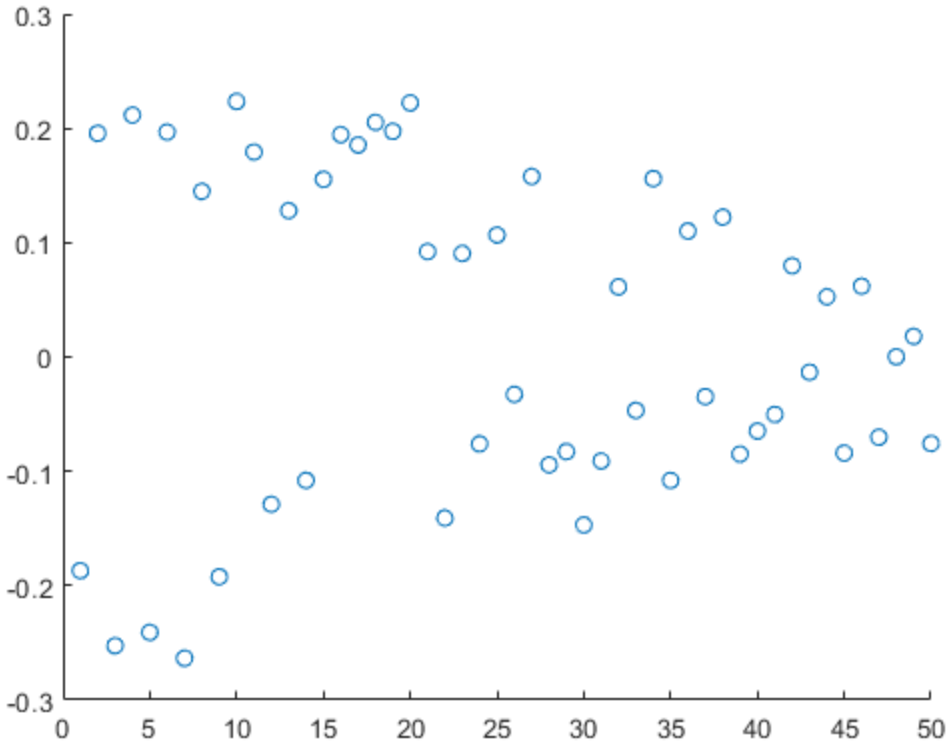
Well check that out! A common question in PCA is "how many components should I keep?" You're looking at an answer: there's a clear drop off after the fifth component, and the variance explained by dimensions 6-50 are all comparably small. We can see this gap even more clearly on a log scale.

```
scatter(1:50,log(var))
```



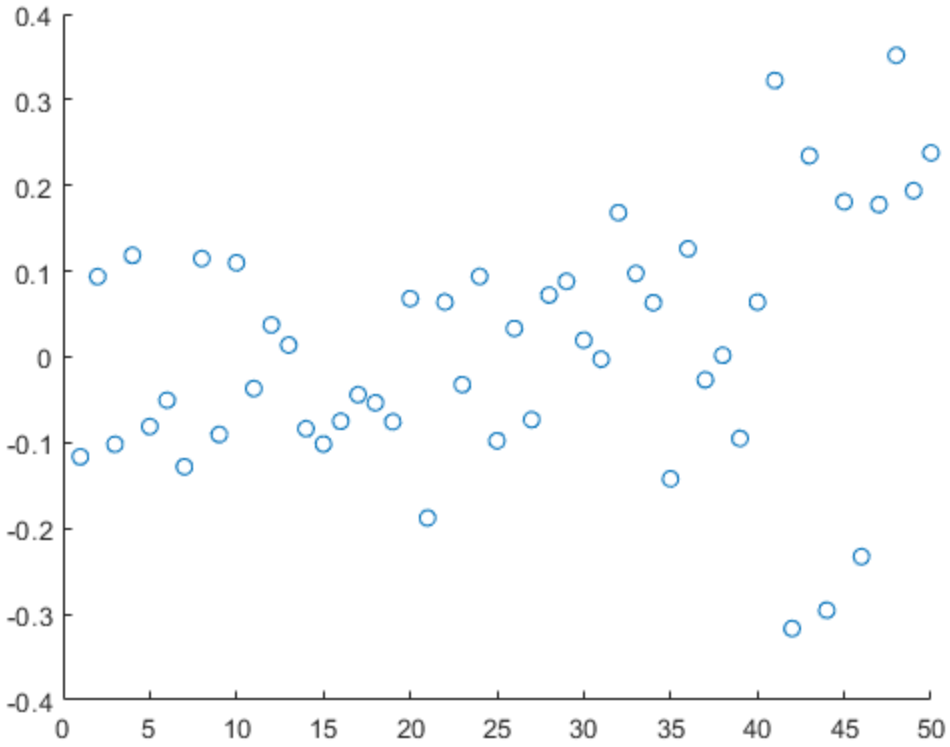
Okay so we're gonna keep five components. Great. Let's look at the first one.

```
scatter(1:50, V(:,1))
```



A problem with PCA, from the social science perspective, is that it looks for the single axis that explains the most possible variance. As a consequence, lots and lots of loadings tend to be non-zero. This first principal axis seems to be more sensitive to the first 20 items than the last 30, but it's messy.

```
scatter(1:50, V(:,4))
```



This axis, the fourth principal component, is much easier to interpret. Its ten largest loadings are on items 41-50. The largest positive loadings are items 41 and 48, and the largest negative loading is item 42. ***There's a codebook for these items in the MATLAB folder. Check it out.*** *41: "I have a rich vocabulary." 48: "I use difficult words." And 42: "I have difficulty understanding abstract ideas." Cool! Taken together, these ten items are coded as "openness to experience." It would be great if we could do a similar thing for the other dimensions, right?*

```
[Rot,T] = rotatefactors(V(:,1:5),'method','varimax');
```

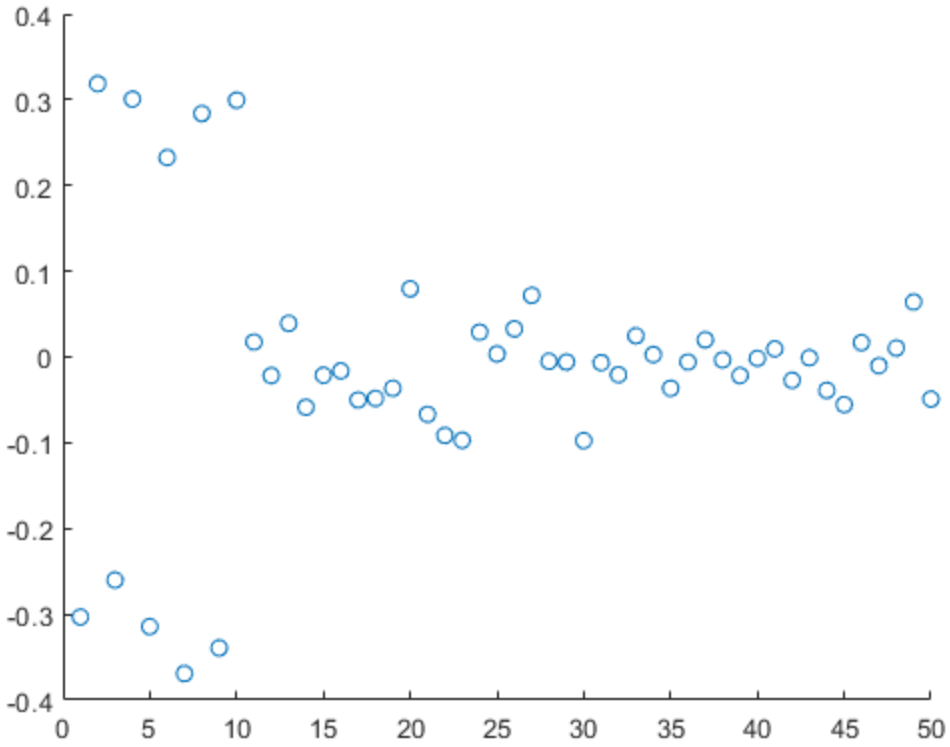
The first five principal components form a 5-dimensional space of the full 50-dimensional feature space that tries to explain as much of the data as possible using only five orthogonal dimensions. A varimax rotation tries to make that five-dimensional space more easily interpretable by humans - it finds a basis for the space that's as close to the original set of axes as possible. ***Prove that Span(Col(Rot)) and Span(Col(V(:,1:5))) are exactly the same subspace of R^50, and that the columns of Rot form an orthogonal basis for this space.*** *The columns of Rot are no longer eigenvectors, so what we're doing is no longer a PCA; it's called a varimax-rotated PCA. MATLAB and R and SPSS also call this a "factor analysis," but that's dangerous. (For more on the difference, you should really take a data analytics course. I am not a statistician.)*

```
Vcol = V(:,1:5);
rot_rr = rref(Rot);
rot_rr(1:5,:)
Rot' * Rot
spantest = horzcat(Vcol,Rot);
result = rref(spantest);
result(1:5,:)
```

```
ans =

     1     0     0     0     0
     0     1     0     0     0
     0     0     1     0     0
     0     0     0     1     0
     0     0     0     0     1


ans =

    1.0000    0.0000   -0.0000   -0.0000   -0.0000
    0.0000    1.0000   -0.0000    0.0000    0.0000
   -0.0000   -0.0000    1.0000   -0.0000    0.0000
   -0.0000    0.0000   -0.0000    1.0000   -0.0000
   -0.0000    0.0000    0.0000   -0.0000    1.0000


ans =

  Columns 1 through 7

    1.0000         0         0         0         0    0.6735    0.5733
         0    1.0000         0         0         0   -0.5750    0.6442
         0         0    1.0000         0         0    0.2239    0.3957
         0         0         0    1.0000         0    0.3266   -0.0958
         0         0         0         0    1.0000   -0.2428    0.3009

  Columns 8 through 10

   -0.2895   -0.1558    0.3310
   -0.4181    0.1082   -0.2605
    0.6570   -0.0665   -0.5976
   -0.2047    0.8790   -0.2640
    0.5175    0.4324    0.6290
```

Col(Rot) and Col(V(:,1:5)) are both 5 dimensional subspaces of R^50. The pivot columns of Rot (in this case, all of them) form a basis for Col(Rot). The matrix Rot multiplied by its transpose is the identity matrix, so this basis is orthogonal. We test if the columns of Rot are in Col(V(:,1:5)) by creating the matrix spantest from V(:,1:5) and Rot and row reducing. The system produced by the row reduction is consistent, so the columns of Rot are in Col(V(:,1:5)). As both Col(Rot) and Col(V(:,1:5)) are the same dimension and the basis elements for Col(Rot) are in Col(V(:,1:5)), the subspaces are in fact the same 5 dimensional subspace of R^50.

```
scatter(1:50,Rot(:,1))
```



Collectively, the five columns of Rot explain exactly as much of the variance as the first five principal components. These rotated columns can no longer be interpreted individually, because they're all super-positions of principal axes, but as a set they have the same explanatory power. Clearly this first rotated axis pulls out the first ten items in the survey. ***Find the questions in the codebook with the largest positive and negative loadings for Rot(:,1), and write them below.*** *This axis is called "extroversion."*

```
largest_pos = max(Rot(:,1)) % Question 2
largest_neg = min(Rot(:,1)) % Question 7
```

```
largest_pos =

    0.3186
```

```
largest_neg =

   -0.3692
```

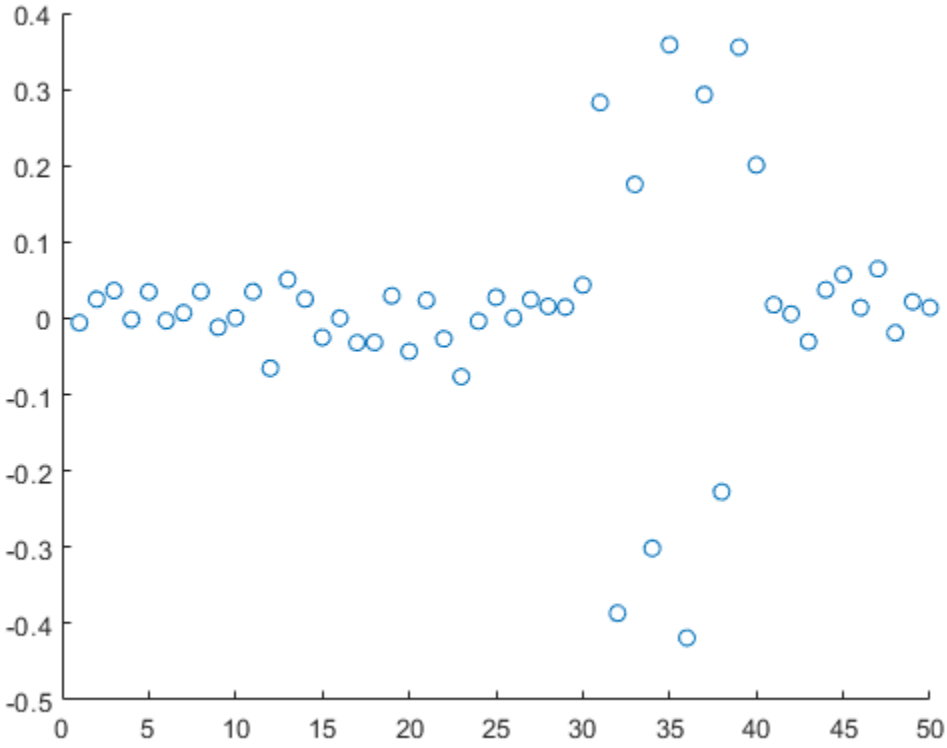
The question with the largest positive loading was "I don't talk a lot". The question with the largest negative loading was "I talk to a lot of different people at parties."

Find the questions with the largest positive and negative loadings for Rot(:,3). *There's nothing special about the order of the items - MATLAB's not looking at that when it does PCA or varimax. This is 100% automatic, it's just what the data is telling us. So the fact that the third column of Rot picks out the "fourth" set of questions is arbitrary; the columns of Rot can only be understood as a set. "Positive" and "negative"*

are also arbitrary - this axis is also upside-down from its usual labeling, which is politely termed the "agreeableness" axis.

```
scatter(1:50,Rot(:,3))
largest_pos2 = max(Rot(:,3)) % Question 35
largest_neg2 = min(Rot(:,3)) % Question 36
```

```
largest_pos2 =

    0.3584


largest_neg2 =

   -0.4187
```



The question with the largest positive loading was "I get chores done right away." The question with the largest negative loading was "I often forget to put things back in their proper place."

I strongly recommend looking at all five of the rotated axes. Social science research is amazing and extremely difficult because humans are complicated: the fact that each of these axes lines up exactly with the traits that it's supposed to is powerful evidence of good survey design. (And why the 1992 paper of Goldberg that developed these items, https://psycnet.apa.org/record/1992-25730-001, has nearly 6000 citations.)

Published with MATLAB® R2019b