Practice Problems based on Week 4 content

General Course Logistics

- Finish your project proposal.

Install Gurobi using the instructions here:

1. Register for an account here.

https://www.gurobi.com/downloads/gurobi-software/

1. Then go back to the same link and download the installer.

2. Apply for a student license:

https://www.gurobi.com/downloads/end-user-license-agreement-academic/

1. After you have successfully gotten the software license, run the command it gives you (it should be a gpg key command or something like that). ***Note: You might need to be on a campus WiFi for this command to be successful. If you're unable to access campus WiFi, connect through Wiscvpn. Directions for installing are here https://it.wisc.edu/services/wiscvpn/***

2. Add the package in Julia (Pkg.add("Gurobi")).

Least Squares

An overdetermined system

Suppose we are measuring a road with four points A;B;C and D. We have the following:

- AD = 89m
- AC = 67m
- BD = 53m
- AB = 35m
- CD = 20m.

We are interested in determining the length of $x_1 = AB$, $x_2 = BC$, and $x_3 = CD$. We have 5 equations and only 3 unknowns. Use least squares to determine what x should be. Are we able to find x such that $Ax = b$?

Autoregressive Models

What if we wanted to do a moving average model, but we didn't have access to the inputs at all? Instead, we areasked to predict future y values based only on the previous y values. One way to do this is by using anautoregressive (AR) model, where each output is approximated by a linear combination of the most recent outputs (excluding the present one):

$$y_t \approx a_1 y_{t-1} + a_2 y_{t-2} + \cdots + a_\ell y_{t-\ell}$$

.

Using the same dataset from class 'moving avg data.csv,' plot the true y, and on the same axes also plot the estimated $\hat{y}$ using the AR model. Use $k = 5$ for the AR model. To quantify the difference between estimates,also compute $||y - \hat{y}||$.

Tradeoff Problems

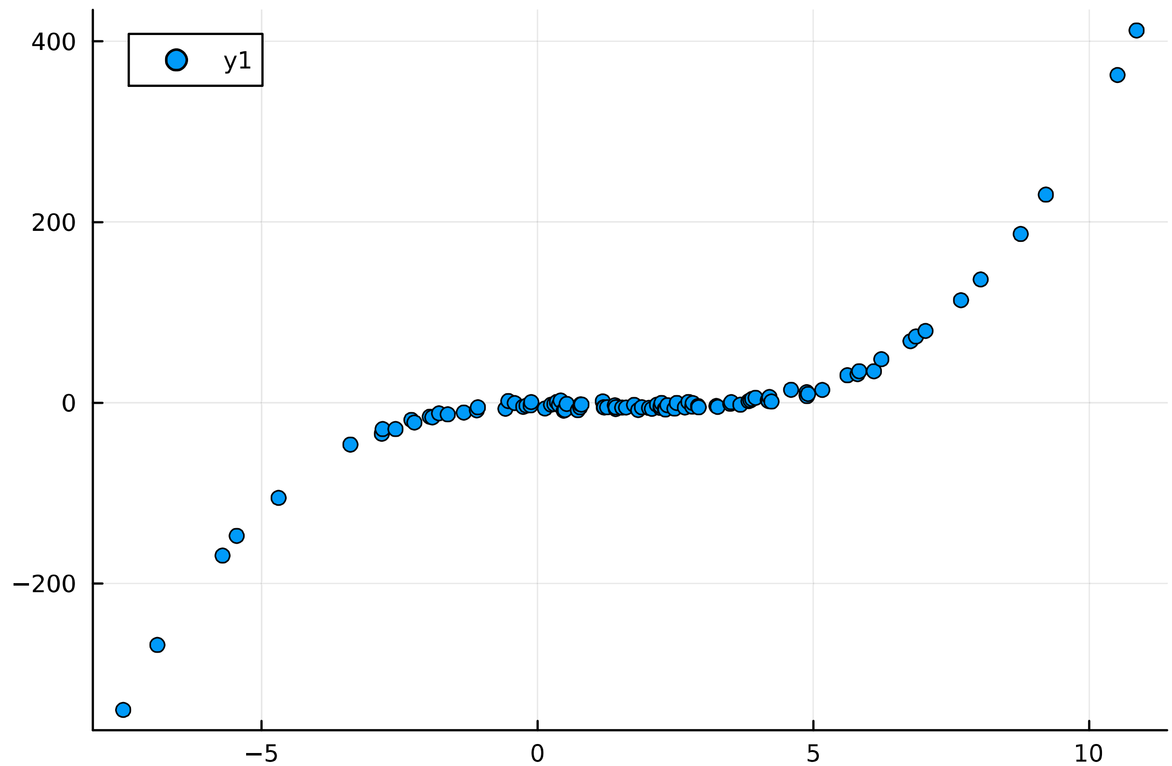
- In lecture, we learned that many constraints are objectives in disguise. For one of the problems we have solved in this course, take a constraint of your choice and move it into the objective with a penalty term. See how your problem behaves differently for different values of the penalty parameter. Try building a Pareto curve.

Regularization

So far in this course, we've seen how to do "standard" polynomial regression by using least squares to fit a curve. However, this is not the only way to fit a model to given data. Consider the following dataset, generated by the code given below (note the data will be different every time you run the code!):

In [6]:
```
using Plots, Distributions
x = 2 .- 3 .* rand(Normal(0, 1),100)
x = sort(x)
y = x .- 2 .* (x.^2) + 0.5 .* (x.^3) + rand(Normal(-3, 3),100)
scatter(x,y,s=:dot)
```

Out[6]:



(a) Perform ordinary polynomial regression on this dataset. Generate plots to show how well the data is fit by a polynomials of degree $d = 5$. Comment on the size of the coefficients in the model you build. Are they large? Small? Sparse? Smooth?

(b) To get a sparser set of coefficients, we'll use Lasso regression (L_1 regularization). Re-sovle the $d = 5$ version of the problem with a regularization parameter of $\lambda = 500$ and plot the new fit. How does the fit change compared to the non-regularized version in part (a)? How do the magnitudes of the coefficients change?

(c) We could also choose to "smooth" the coefficients by using Ridge regression (L_2 regularization). Re-sovle the $d = 5$ version of the problem with a regularization parameter of $\lambda = 500$ and plot the new fit. How does the fit change compared to the non-regularized version in part (a) and the L_1 regularization in part (b)? How do the magnitudes of the coefficients change? Calculate the 2-norm of the solution you obtained in (a) and compare it to the 2-norm of the solution in (c).

In [ ]: