Problem Set #1

John Nash
AEB6933 - Fantastic Econometric Course
University of Florida

April 26, 2020

Question 1

You can write the questions here ...
And your answers here ...

Question 1.a

Also, you can write your sub-questions into subsection levels.
And again, your answers here ...

Mathematical notation

This template offers a set of customized mathematical symbols commonly used in FRED courses.

You can invoke the following symbols:

```
\E(\cdot)             % Expectation
\V(\cdot)             % Variance
\Var(\cdot)           % Variance
\Cov(\cdot)           % Covariance
\Corr(\cdot)          % Correlation
\tr(\cdot)            % Trace
\rank(\cdot)          % Rank
\N(0,1)               % Normal
\op(\cdot)            % "little o" ope
\Op(\cdot)            % "big o" operator
\R^+                  % Real positive num
\La(y|\beta,\sigma^2) % Lagrange function
```

- $\mathrm{E}(\cdot)$
- $\mathrm{V}(\cdot)$
- $\mathrm{Var}(\cdot)$
- $\mathrm{Cov}(\cdot)$
- $\mathrm{Corr}(\cdot)$
- $\mathrm{tr}(\cdot)$
- $\mathrm{rank}(\cdot)$
- $\mathcal{N}(0,1)$
- $o(\cdot)$
- $\mathcal{O}(\cdot)$
- $\mathbb{R}^+$
- $\mathcal{L}(y|\beta,\sigma^2)$

Remember, these symbols have to be implemented into equation environments, that means, with dollar symbols $…$, or `equation`-environments.

There is a set of different accents you may use:

```
$\hat \beta$, $\widehat \beta$, $\tilde \alpha$, $\widetilde \alpha$, $\bar y$,
$\varepsilon \sim \chi^2_t$
```

Which reproduce the following: $\hat \beta$, $\widehat \beta$, $\tilde \alpha$, $\widetilde \alpha$, $\bar y$, $\varepsilon \sim \chi^2_t$.
These are a couple of examples:

$$\mathrm{Avar}(b) = \frac{\sigma^2}{n} Q^{-1} \operatorname{plim} \left(\frac{1}{n} X' Q X \right) Q^{-1} \tag{1}$$

$$\sqrt{n}(b - \beta) \xrightarrow{d} \mathcal{N} \left[0, \frac{\sigma^2}{n} Q^{-1} \operatorname{plim} \left(\frac{1}{n} X' \Omega X \right) Q^{-1} \right] \tag{2}$$

Code and scripts

If you want to print the raw outcomes from any software, it is recommended the `Verbatim`-environment :

```
Parameters     Estimates      Std. err.  Est./s.e.  Prob.    Gradient
-------------------------------------------------------------------
P01              -0.5553        0.1446    -3.840   0.0001      0.0002
P02              -0.2255        0.1106    -2.039   0.0414     -0.0004
P03              -0.8588        0.2378    -3.612   0.0003     -0.0003
```

The same environment can be used for print a chunk of code (this is GAUSS by the way):

```
ev = ev1 ~ ev2;
ev1 = sumc(ev[.,nest1]');
ev2 = sumc(ev[.,nest2]');
num = (ev1 .^ (k[1]-1)).*sumc(depm[.,nest1]') + (ev2 .^ (k[2]-1)).*sumc(depm[.,nest2]');
p = sumc((ev .* depm)').* num ./ ((ev1.^k[1])+(ev2.^k[2]));
```

If you want to print Stata-code, you can use the environment `lstlisting`, setting the `style` in `stata-editor`. This will highlight the Stata commands like the following example:

```
clogit depvar var1 var2 var3, group(id)
```

The same `lstlisting`-package can be used to print **R**-code, setting the only argument (`language`) in `R`.

```
Quad.reg <- function(z) {
  grad <- cbind(z[4]+z[11]*m[1]+z[15]*m[2]+z[25]*m[3]+z[19]*m[4]+z[20]*m[5]+z[21]*m[6],
                z[5]+z[12]*m[1]+z[16]*m[2]+z[19]*m[3]+z[26]*m[4]+z[22]*m[5]+z[23]*m[6],
                z[7]+z[14]*m[1]+z[18]*m[2]+z[21]*m[3]+z[23]*m[4]+z[22]*m[5]+z[28]*m[6])
  test1 <- min(grad) # have to be >0
  test2 <- max(eigen(hess)$values) # have to be < 0
  if(Mode == 1) {
    lst <- list(Grad = grad, Hess = hess, Test1 = test1, Test2 = test2)
    return(lst)
  } else {
    return(cbind(test1,test2))
  }
}
```