Econ 675: HW 3

Erin Markiewitz

October 29, 2018

Contents

1 Non-linear Least Squares

1.1

The estimator $\beta_{\mathbf{0}} = \arg\min_{\beta \in \mathbb{R}^d} \mathbb{E}[(y_i - \mu(\mathbf{x_i'}\beta))^2]$ is identified if the following condition must be met:

$$\begin{aligned}
\mathbb{E}[y_i - \mu(\mathbf{x_i'}\beta))^2] &= \mathbb{E}[(y_i - \mu(\mathbf{x_i'}\beta_{\mathbf{0}}) + \mu(\mathbf{x_i'}\beta_{\mathbf{0}}) - \mu(\mathbf{x_i'}\beta))^2] \\
&= \mathbb{E}[(y_i - \mu(\mathbf{x_i'}\beta_{\mathbf{0}}))^2 + (\mu(\mathbf{x_i'}\beta_{\mathbf{0}}) - \mu(\mathbf{x_i'}\beta))^2] + 2\mathbb{E}[(y_i - \mu(\mathbf{x_i'}\beta_{\mathbf{0}}))(\mu(\mathbf{x_i'}\beta_{\mathbf{0}}) - \mu(\mathbf{x_i'}\beta))]
\end{aligned}$$

the cross term is zero by the law of iterated expectations since

$$\begin{aligned}
\mathbb{E}[(y_i - \mu(\mathbf{x_i'}\beta_{\mathbf{0}}))(\mu(\mathbf{x_i'}\beta_{\mathbf{0}}) - \mu(\mathbf{x_i'}\beta))] &= \mathbb{E}[y_i\mu(\mathbf{x_i'}\beta_{\mathbf{0}}) - \mu(\mathbf{x_i'}\beta_{\mathbf{0}})^2 + \mu(\mathbf{x_i'}\beta_{\mathbf{0}})\mu(\mathbf{x_i'}\beta)) - y_i\mu(\mathbf{x_i'}\beta))] \\
&= \mathbb{E}[\mu(\mathbf{x_i'}\beta_{\mathbf{0}})^2 - \mu(\mathbf{x_i'}\beta_{\mathbf{0}})^2 + \mu(\mathbf{x_i'}\beta_{\mathbf{0}})\mu(\mathbf{x_i'}\beta)) - \mu(\mathbf{x_i'}\beta_{\mathbf{0}})\mu(\mathbf{x_i'}\beta))] \\
&= 0
\end{aligned}$$

Now we can rewrite the previous expression, iterating expectations again

$$\mathbb{E}[(y_i - \mu(\mathbf{x_i'}\beta_{\mathbf{0}}))^2 + (y_i - \mu(\mathbf{x_i'}\beta))^2] = \mathbb{E}[0 + (y_i - \mu(\mathbf{x_i'}\beta))^2]$$

Thus, if

$$\mathbb{E}[(y_i - \mu(\mathbf{x_i'}\beta))^2] >= \mathbb{E}[(y_i - \mu(\mathbf{x_i'}\beta_{\mathbf{0}}))^2], \quad \forall \beta \neq \beta_0$$

then β_0 is identified.

1.2

To prove convergence in distribution we need take the first order condition of the finite sample analogue:

$$\hat{\beta}_n = \arg\min_{\beta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} (y_i - \mu(\mathbf{x_i'}\beta))^2$$

F.O.C.

$$0 = \frac{1}{n} \sum_{i=1}^{n} (y_i - \mu(\mathbf{x_i'}\beta))\dot{\mu}(x_i'\beta)x_i \equiv \frac{1}{n} \sum_{i=1}^{n} m(z_i, \hat{\beta}_n)$$

Sufficient conditions for convergence in distribution are 1) uniform consistency so $\hat{\beta}_n \to_p \beta_0$ and 2) regularity conditions of the $m(.,.)$ functions (twice differentiable, integrable second derivatives, finite variance, invertibility of the first derivative). All of these regularity conditions allow us to take a first-order taylor expansion of the m function around β_0.If we have all of that the estimator converges in distribution to:

$$\begin{aligned} 0 &= \frac{1}{n}\sum_{i=1}^{n}\mathbf{m(z_i, \beta_0)} + \frac{1}{n}\sum_{i=1}^{n}\dot{m}(z_i, \beta_0)(\hat{\beta}_n - \beta_0) \\ \sqrt{n}(\hat{\beta}_n - \beta_0) &= \left(\frac{1}{n}\sum_{i=1}^{n}\dot{m}(z_i, \beta_0)\right)^{-1}\frac{1}{\sqrt{n}}\sum_{i=1}^{n}\mathbf{m(z_i, \beta_0)} \end{aligned}$$

So the estimator converges in distribution to:

$$\sqrt(n)(\hat{\beta}_n - \beta_0) \to_d N(0, H_0^{-1}\Sigma_0 H_0^{-1})$$

where

$$H_0 = \mathbb{E}[\dot{m}(z_i, \beta_0)] = \mathbb{E}[\dot{\mu}(x_i'\beta_0))^2 x_i' x_i]$$

and

$$\begin{aligned} \Sigma_0 = \mathbb{V}[\mathbf{m(z_i, \beta_0)}] &= \mathbb{V}[(y_i - \mu(\mathbf{x_i'}\beta_\mathbf{0}))\dot{\mu}(x_i'\beta_0)x_i] \\ &= \mathbb{E}[(y_i - \mu(\mathbf{x_i'}\beta_\mathbf{0}))^2\dot{\mu}(\mathbf{x_i'}\beta_\mathbf{0})^\mathbf{2}\mathbf{x_i'x_i}] \\ &= \mathbb{E}[\mathbb{E}[(y_i - \mu(\mathbf{x_i'}\beta_\mathbf{0}))^2\dot{\mu}(\mathbf{x_i'}\beta_\mathbf{0})^\mathbf{2}\mathbf{x_i'x_i}|x_i] \\ &= \mathbb{E}[\mathbb{E}[(y_i - \mu(\mathbf{x_i'}\beta_\mathbf{0}))^2|x_i]\dot{\mu}(\mathbf{x_i'}\beta_\mathbf{0})^\mathbf{2}\mathbf{x_i'x_i}] \\ &= \mathbb{E}[\sigma(x_i)\dot{\mu}(\mathbf{x_i'}\beta_\mathbf{0})^\mathbf{2}\mathbf{x_i'x_i}] \end{aligned}$$

1.3

Now as

$$V_0 = H_0^{-1}\Sigma_0 H_0^{-1} = \mathbb{E}[\sigma(x_i)(\dot{\mu}(\mathbf{x_i'}\beta_\mathbf{0})^\mathbf{2}\mathbf{x_i'x_i})^{-1}]$$

we have a heteroskedastic consistent variance estimator

$$\hat{V}_n^{HC} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \mu(x_i'\hat{\beta}_n))^2(\dot{\mu}(x_i'\hat{\beta}_n)^2 x_i' x_i)^{-1}$$

As long as $\hat{\beta}_n \to_p \beta_0$, by the delta method we can construct a asymptotic variance for inference. For the delta method, let $g(x) = ||\beta||^2 = \sum_{i=1}^{d}\beta^i$, so $\dot{g}(x) = 2\beta'$ so

$$\sqrt{n}(||\hat{\beta}_n||^2 - ||\beta_0||^2) \to_d N(0, 4\beta_0' H_0^{-1} \Sigma_0 H_0^{-1} \beta_0)$$

so we can construct a 95% confidence interval for $||\hat{\beta}_n||^2$

$$CI_{95}\left(||\hat{\beta}_n||^2\right) = \left[||\hat{\beta}_n||^2 - 1.96\sqrt{4\hat{\beta}_n' \hat{V}_n^{HO} \hat{\beta}_n'/n}\ ,\ ||\hat{\beta}_n||^2 + 1.96\sqrt{4\hat{\beta}_n' \hat{V}_n^{HO} \hat{\beta}_n'/n}\right]$$

1.4

If $\sigma(x_i) = \sigma$ then V_0 simplifies to $V_0 = \sigma H^{-1}$. So to estimate V_0 all we need to do is put hats on things: $\hat{V}_0^{HO} = \hat{\sigma}\hat{H}^{-1}$. where

$$\hat{\sigma} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \mu(x_i'\hat{\beta}_n))^2$$

and

$$\hat{H} = \frac{1}{n}\sum_{i=1}^{n}\dot{\mu}(x_i'\hat{\beta}_n)^2 x_i' x_i$$

As long as $\hat{\beta}_n \to_p \beta_0$, by the delta method we can construct a asymptotic variance for inference. For the delta method, let $g(x) = ||\beta||^2 = \sum_{i=1}^{d} \beta^i$, so $\dot{g}(x) = 2\beta'$ so

$$\sqrt{n}(||\hat{\beta}_n||^2 - ||\beta_0||^2) \to_d N(0, 4\beta_0' H_0^{-1} \Sigma_0 H_0^{-1} \beta_0)$$

so we can construct a 95% confidence interval for $||\hat{\beta}_n||^2$

$$CI_{95}\left(||\hat{\beta}_n||^2\right) = \left[||\hat{\beta}_n||^2 - 1.96\sqrt{4\hat{\beta}_n' \hat{V}_n^{HO} \hat{\beta}_n'/n}\ ,\ ||\hat{\beta}_n||^2 + 1.96\sqrt{4\hat{\beta}_n' \hat{V}_n^{HO} \hat{\beta}_n'/n}\right]$$

1.5

We start with the log-likelihood function and take first order conditions

$$\hat{\beta}_{ML} = \arg\min_{\beta \in \mathbb{R}^d} -log((2pi)^{\frac{n}{2}}\sigma^n) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \mu(\mathbf{x_i'}\beta))^2 - \frac{n}{2}log(\sigma^2)$$

$$
\begin{aligned}
\partial\beta: \ & \frac{1}{\hat{\sigma}^2_{ML}} \sum_{i=1}^{n} (y_i - \mu(x_i'\hat{\beta}_{ML}))\dot{\mu}(x_i'\hat{\beta}_{ML})x_i = 0 \\
\partial\sigma^2: \ & \frac{1}{\hat{\sigma}^4_{ML}} \sum_{i=1}^{n} (y_i - \mu(x_i'\hat{\beta}_{ML}))^2 - \frac{n}{2\hat{\sigma}^2_{ML}} = 0
\end{aligned}
$$

The first FOC implies $\hat{\beta}_{ML} = \hat{\beta}_n$ the second FOC provides the same variance estimator as the previous question. Therefore the estimator coincides with the one in the previous secion.

1.6

If the link function is unkown, β_0 is not identifiable as there are infintely many pairs of parameters and functions that can minimize the original least squares objective function. For instance let $\mu^A(x_i'\beta_A) = x_i'\beta_0$ and $\mu^B(x_i'(\beta_B)) = \mu^B(x_i'(\beta_0)^{-1}) = x_i'\beta_0$. You can restore identifiability by assuming $||\beta_0|| = 1$

1.7

Ok so the link function is

$$
\begin{aligned}
\mu^B(x_i'(\beta_0)^{-1}) &= \mathbb{E}[y_i|x_i] \\
&= \mathbb{E}[1(x_i'\beta_0 - \epsilon_i \geq 0)] \\
&= \mathbb{E}[1(x_i'\beta_0 - \epsilon_i \geq 0)|x_i] \\
&= Pr(x_i'\beta_0 \geq \mathbb{E}[\epsilon_i|x_i]) \\
&= \frac{1}{1 + \exp(-x_i'\beta)}, \text{if } s_0 = 1
\end{aligned}
$$

So the link function is the inverse of the logistic c.d.f.. Next we can derive the formula of the conditional variance of x_i, $\sigma^2(x_i)\mathbb{V}[y_i|x_i]$

Since $y_i|x_i \sim Bernoulli(F(x_i'\beta_0))$

$$
\begin{aligned}
\sigma^2(x_i) &= F(x_i'\beta_0)(1 - F(x_i'\beta_0)) \\
&= \mu(\mathbf{x_i'}\boldsymbol{\beta_0})(1 - \mu(\mathbf{x_i'}\boldsymbol{\beta_0}))
\end{aligned}
$$

Then by previous result,

$$
V_0 = H_0^{-1}\Sigma_0 H_0^{-1}
$$

where

$$H_0 = \mathbb{E}[(1-\mu(\mathbf{x_i'}\beta_\mathbf{0}))^2\mu(\mathbf{x_i'}\beta_\mathbf{0})^2 x_i x_i]$$

and

$$\Sigma_0 = \mathbb{E}[(1-\mu(\mathbf{x_i'}\beta_\mathbf{0}))^3\mu(\mathbf{x_i'}\beta_\mathbf{0})^3 x_i x_i]$$

as $\dot{\mu}(x) = (1-\mu(u))\mu(u)$

1.8

By previous resultm MLE will give the same point estimate as NLS, but $V_{NLS} \geq V_{MLE}$ as MLE is asymptotically efficient.

1.9

1.9.1

Stata output:

	$\hat{\beta}_n$	$\hat{\mathbf{V}}_n^{HC}$	tstat	pvalue	CI_{95}
S_age	1.333361	.0151533	10.83165	0	1.092092,1.57463
S_HHpeople	-.0665942	.0005378	-2.871698	.0040827	-.1120454,-.0211429
ls_incomepc	-.118689	.0019204	-2.708397	.0067609	-.2045796,-.0327983
Constant	1.755024	.1118828	5.246883	1.55e-07	1.099438,2.41061

R output:

	$\hat{\beta}_n$	$\hat{\mathbf{V}}_n^{HC}$	tstat	pvalue	CI_{95} lower	CI_{95} Upper
S_age	1.33336	0.01517	10.82627	0.00000	1.09197	1.57475
S_HHpeople	-0.06659	0.00054	-2.87061	0.00400	-0.11206	-0.02113
log_inc	-0.11869	0.00192	-2.70742	0.00700	-0.20461	-0.03277
Constant	1.75502	0.11197	5.24491	0.00000	1.09919	2.41086

1.9.2

Stata output:

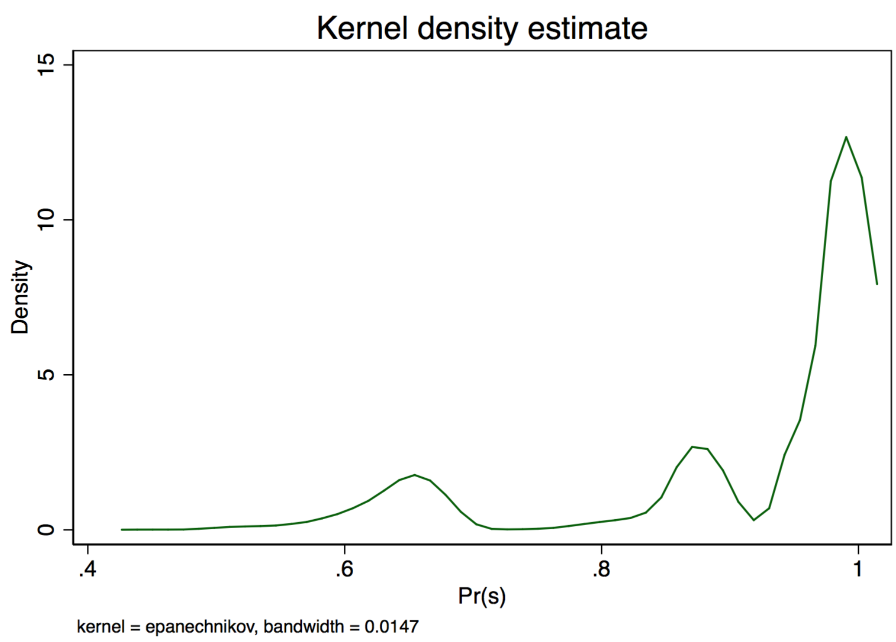
	$\hat{\beta}_n$	$\hat{\mathbf{V}}_n^{HC}$	tstat	pvalue	CI_{95}
S_age	1.333361	.0151859	10.82	0	1.091832,1.57489
S_HHpeople	-.0665942	.0005547	-2.827487	.0046915	-.1127561,-.0204323
ls_incomepc	-.118689	.0020023	-2.652454	.0079909	-.2063912,-.0309867
Constant	1.755024	.1267694	4.929193	0	1.057185,2.452863

R output:

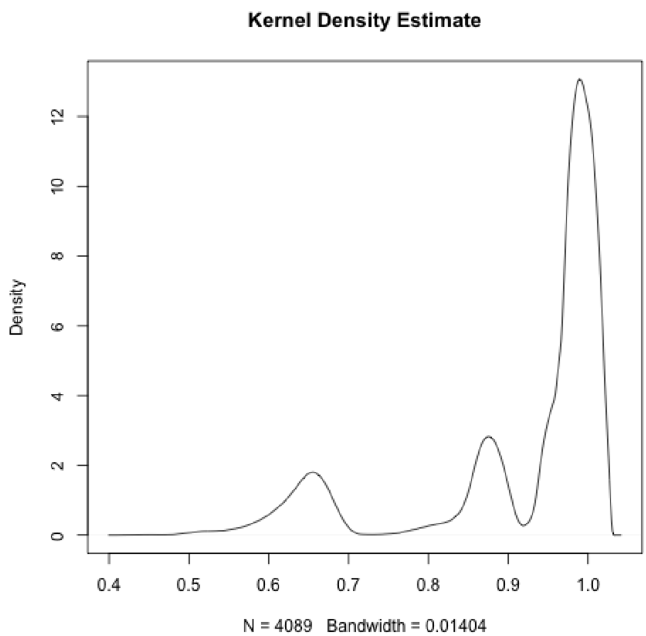
	$\hat{\beta}_n$	CI_{95} lower	CI_{95} Upper	pvalue
S_age	1.33	1.16	1.85	0.00
S_HHpeople	-0.07	-0.13	-0.01	0.00
ls_incomepc	-0.12	-0.31	-0.05	0.00
Constant	1.76	1.13	3.21	0.00

1.9.3

Stata output:



R output:



Kernel Density Estimate
Density
0
2
4
6
8
10
12
0.4
0.5
0.6
0.7
0.8
0.9
1.0
N = 4089 Bandwidth = 0.01404

2 Semiparametric GMM with Missing Data

2.1

2.1.1

Consider the following moment condition of a GMM estimator:

$$\mathbb{E}[m(\mathbf{y_i^*}, \mathbf{t_i}, \mathbf{x_i}; \beta_\mathbf{0})|t_i, x_i] = 0$$

by the law of iterated expectations, the following conditions hold as well.

$$\begin{aligned}
\mathbb{E}[g(\mathbf{t_i}, \mathbf{x_i})\mathbb{E}[m(\mathbf{y_i^*}, \mathbf{t_i}, \mathbf{x_i}; \beta_\mathbf{0})|t_i, x_i] &= 0 \\
\mathbb{E}[\mathbb{E}[g(\mathbf{t_i}, \mathbf{x_i})m(\mathbf{y_i^*}, \mathbf{t_i}, \mathbf{x_i}; \beta_\mathbf{0})|t_i, x_i]] &= 0 \\
\mathbb{E}[g(\mathbf{t_i}, \mathbf{x_i})m(\mathbf{y_i^*}, \mathbf{t_i}, \mathbf{x_i}; \beta_\mathbf{0})|t_i, x_i] &= 0, \ \forall g(\mathbf{t_i}, \mathbf{x_i})
\end{aligned}$$

In order to find the function $g_0(\mathbf{t_i}, \mathbf{x_i})$ that minimizes asymptotic variance of the estimator, we write down the objective function and take first order conditions.

$$\hat{\beta} = \arg\min_{\hat{\beta}} \left(\frac{1}{n}\sum_{i=1}^n g(\mathbf{t_i}, \mathbf{x_i})m(\mathbf{y_i^*}, \mathbf{t_i}, \mathbf{x_i}; \hat{\beta})\right)' W \left(\frac{1}{n}\sum_{i=1}^n g(\mathbf{t_i}, \mathbf{x_i})m(\mathbf{y_i^*}, \mathbf{t_i}, \mathbf{x_i}; \hat{\beta})\right)$$

F.O.C.

$$0 = \left(\frac{1}{n}\sum_{i=1}^n \frac{\partial}{\partial\beta} g(\mathbf{t_i}, \mathbf{x_i})m(\mathbf{y_i^*}, \mathbf{t_i}, \mathbf{x_i}; \hat{\beta})\right)' W \left(\frac{1}{n}\sum_{i=1}^n \frac{\partial}{\partial\beta} g(\mathbf{t_i}, \mathbf{x_i})m(\mathbf{y_i^*}, \mathbf{t_i}, \mathbf{x_i}; \hat{\beta})\right)$$

Next we take a first order taylor expansion of the m function around β_0:

$$\begin{aligned}
0 = &\left(\frac{1}{n}\sum_{i=1}^n \frac{\partial}{\partial\beta} g(\mathbf{t_i}, \mathbf{x_i})m(\mathbf{y_i^*}, \mathbf{t_i}, \mathbf{x_i}; \beta_\mathbf{0})\right)' W \left(\frac{1}{n}\sum_{i=1}^n \frac{\partial}{\partial\beta} g(\mathbf{t_i}, \mathbf{x_i})m(\mathbf{y_i^*}, \mathbf{t_i}, \mathbf{x_i}; \beta_\mathbf{0})\right) \\
+ &\left(\frac{1}{n}\sum_{i=1}^n \frac{\partial}{\partial\beta} g(\mathbf{t_i}, \mathbf{x_i})m(\mathbf{y_i^*}, \mathbf{t_i}, \mathbf{x_i}; \beta_\mathbf{0})\right)' W \left(\frac{1}{n}\sum_{i=1}^n \frac{\partial}{\partial\beta} g(\mathbf{t_i}, \mathbf{x_i})m(\mathbf{y_i^*}, \mathbf{t_i}, \mathbf{x_i}; \beta_\mathbf{0})\right)(\hat{\beta} - \beta_0)
\end{aligned}$$

and rearrange and multiply be $\sqrt{n}$ to give us the influence function

$$\sqrt{n}(\hat{\beta} - \beta_0) = \left(\mathbf{\Omega_0' W \Omega_0}\right)^{-1} \Omega_0 W_0 \frac{1}{\sqrt{n}}\sum_{i=1}^n \frac{\partial}{\partial\beta} g(\mathbf{t_i}, \mathbf{x_i})m(\mathbf{y_i^*}, \mathbf{t_i}, \mathbf{x_i}; \beta_\mathbf{0}) + o_p(1)$$

So by the CLT, assuming finite mean and variance of the estimator

$$\sqrt{n}(\hat{\beta}-\beta_0) \rightarrow_d N(0, V_0)$$

where

$$V_0 = \left(\mathbf{\Omega_0'} \mathbf{W} \mathbf{\Omega_0}\right)^{-1} \mathbf{\Omega_0} \mathbf{W} \mathbf{\Sigma_0} \mathbf{W} \mathbf{\Omega_0} \left(\mathbf{\Omega_0'} \mathbf{W} \mathbf{\Omega_0}\right)^{-1}$$

and

$$\Sigma_0 = \mathbb{V}[g(\mathbf{t_i}, \mathbf{x_i}) m(\mathbf{y_i^*}, \mathbf{t_i}, \mathbf{x_i}; \beta_{\mathbf{0}})]$$

Thus, the asymptotic variance is minimized when

$$\mathbf{W}^* = \mathbf{\Sigma_0^{-1}} \tag{1}$$

and

$$g^*(\mathbf{t_i}, \mathbf{x_i}) = \frac{\partial m_i}{\partial \beta} \mathbb{V}[m(\mathbf{y_i^*}, \mathbf{t_i}, \mathbf{x_i}; \beta_{\mathbf{0}}) | t_i, x_i]^{-1}$$

which implies that $V_0^* = \left(\Omega_0' \Sigma_0^{-1} \Omega_0\right)^{-1}$
Cool? Cool. Ok now we apply our findings to the model specified by the question.

$$\mathbb{V}[m(\mathbf{y_i^*}, \mathbf{t_i}, \mathbf{x_i}; \beta_{\mathbf{0}}) | t_i, x_i] = F(t_i\theta_0 + x_i\gamma_0)(1 - F(t_i\theta_0 + x_i\gamma_0))$$

and

$$\mathbb{E}[\frac{\partial m_i}{\partial \beta} | t_i, x_i] = f(t_i\theta_0 + x_i * \gamma_0)[t_i, x_i']'$$

which gives us our result

$$g_0(\mathbf{t_i}, \mathbf{x_i}) = \frac{f(t_i\theta_0 + x_i\gamma_0)}{F(t_i\theta_0 + x_i\gamma_0)(1 - F(t_i\theta_0 + x_i\gamma_0))}[t_i, x_i']'$$

Now if the link function is the logistic cdf, then
$F(x) = \frac{1}{1+\exp(-x)}$ and $f(x) = \frac{-\exp(-x)}{(1+\exp(-x))^2} = -\exp(-x)F(x)^2$. So

$$\begin{aligned}
\frac{f(t_i\theta_0 + x_i\gamma_0)}{F(t_i\theta_0 + x_i\gamma_0)(1 - F(t_i\theta_0 + x_i\gamma_0))}[t_i, x_i']' &= \frac{-\exp(-x)F(x)^2}{(F(x))(1 - F(x))} \\
&= \frac{-\exp(-x)F(x)}{1 - F(x)} \\
&= 1
\end{aligned}$$

gives us $g_0(\mathbf{t_i}, \mathbf{x_i}) = [t_i, x_i']'$

2.2

2.2.1

Using the previous result, the optimal moment condition is

$$\mathbb{E}[g(\mathbf{t_i}, \mathbf{x_i})m(y_i^*, t_i, x_i; \beta_0)] = 0$$

As the outcome variable is missing at completely random, $s_i \perp (y_i^*, t_i, x_i; \beta_0)$

$$\mathbb{E}[g_0(\mathbf{t_i}, \mathbf{x_i})m(y_i^*, t_i, x_i; \beta_0)] = 0$$
$$\mathbb{E}[g_0(\mathbf{t_i}, \mathbf{x_i})m(y_i^*, t_i, x_i; \beta_0)|s_i = 1] = 0$$

Thus, the infesible estimator

$$\hat{\beta}_{MCAR} = \arg\min_{\hat{\beta}_{MCAR}} \left| \hat{\mathbb{E}}[g_0(\mathbf{t_i}, \mathbf{x_i})m(y_i^*, t_i, x_i; \hat{\beta}_{MCAR})|s_i = 1] \right|$$

is a consistent estimator of β_0, and we can construct the feasible estimator

$$\hat{\beta}_{MCAR,feasible} = \arg\min_{\hat{\beta}_{MCAR,feasible}} \left| \hat{\mathbb{E}}[\hat{g}(\mathbf{t_i}, \mathbf{x_i})m(y_i^*, t_i, x_i; \hat{\beta}_{MCAR})|s_i = 1] \right|$$

2.2.2

Stata output:

	$\hat{\beta}_{MCAR}$	CI_{95}
dpisofirme	-.3163832	-.4476255,-.1851408
S_age	-.244022	-.282266,-.2057781
S_HHpeople	.023667	.0009192,.0464149
ls_incomepc	.0325661	.0073571,.057775

R output:

	$\hat{\beta}_{MCAR}$	CI_{95} lower	CI_{95} upper
dpisofirme	-0.33	-0.52	-0.17
S_age	-0.23	-0.27	-0.18
S_HHpeople	0.03	-0.01	0.06
log_inc	0.02	-0.01	0.06

2.3

2.3.1

2.3.2

2.3.3

Stata output:

	$\hat{\beta}_{MCAR}$	CI_{95}
S_age	-.2446852	-.2850008,-.2043697
S_HHpeople	.0241638	-.0019753,.050303
ls_incomepc	.0324512	.0051728,.0597295
dpisofirme	-.315488	-.4492407,-.1817353

R output:

	$\hat{\beta}_{MCAR}$	CI_{95} lower	CI_{95} upper
dpisofirme	-0.32	-0.49	-0.16
S_age	-0.22	-0.27	-0.17
S_HHpeople	0.03	-0.01	0.06
log_inc	0.02	-0.01	0.06

2.3.4

Stata output:

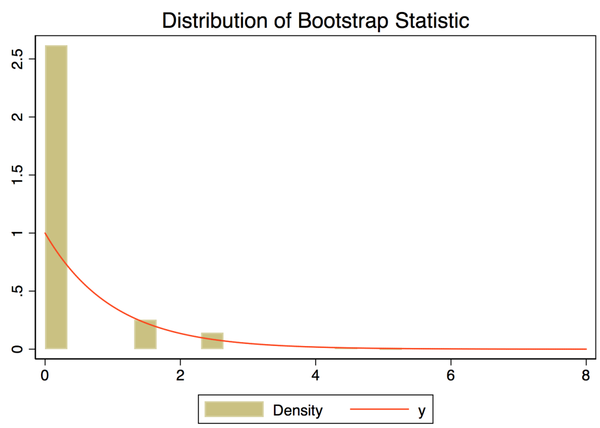
	$\hat{\beta}_{MAR}$	CI_{95}
S_age	-.2446852	-.2822807,-.2070897
S_HHpeople	.0241638	.0000545,.0482732
ls_incomepc	.0324512	.0068349,.0580674
dpisofirme	-.315488	-.4485709,-.1824051

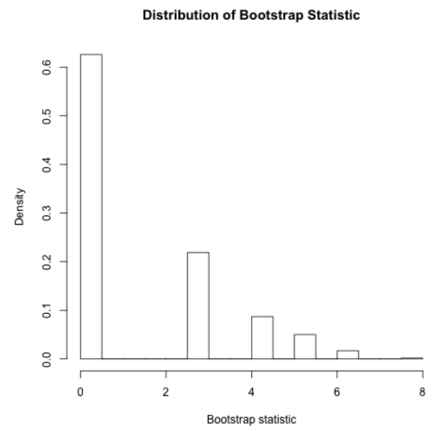
R output:

	$\hat{\beta}_{MAR}$	CI_{95} lower	CI_{95} upper
dpisofirme	-0.32	-0.49	-0.16
S_age	-0.22	-0.27	-0.17
S_HHpeople	0.03	-0.01	0.06
log_inc	0.02	-0.01	0.06

3 When Bootstrap Fails

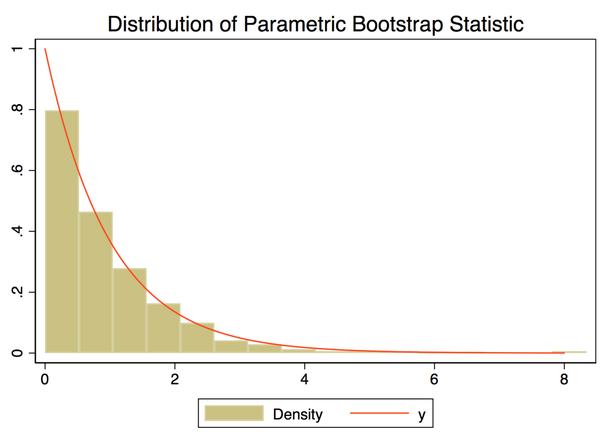
3.1

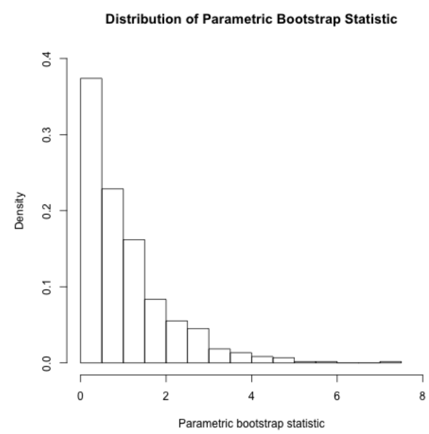




No, it does not coincide with the theoretical Exponential(1) distribution

3.2





Yes, it does coincide with the theoretical Exponential(1) distribution

3.3

The intuitive reason behind why the nonparametic bootsstrap fails is that by this method $\mathbb{E}[\max_i x_i^*] = \frac{2}{n}\sum_{i=1}^{n} x_i \neq \max_i x_i$.

While in the case of the parametric it works out, as $\mathbb{E}[\max_i x_i^*] = \max_i x_i$ by construction

4 Code Appendix

Stata

```
// Erin Markiewitz
// ECON 675 Assignment 3
********************************************************************************
clear all
set more off, perm
set seed 12345
global dir "/Users/erinmarkiewitz/Dropbox/Phd_Coursework/Econ675/hw3"
global datadir $dir\data
global resdir $dir\results

cap log close
log using $resdir\pset2_stata.smcl, replace



*******
*** Problem 1
*******
use pisofirme, clear
gen s = 1 - cond(danemia==.,1,0)
gen ls_incomepc =log(S_incomepc+1)
glm s S_age S_HHpeople ls_incomepc, family(binomial) link(logit) r
estout using hw3_q1_9a_stata.tex, cells("b var t p ci") style(tex)  replace



glm s S_age S_HHpeople ls_incomepc, family(binomial) link(logit) vce(bs, r(99)  seed(123) nodots)
estout using hw3_q1_9b_stata.tex, cells("b p ci") style(tex)  replace
predict prop_score
kdensity prop_score
gr export  hw3_q1_9a_stata.png, replace



*******
*** Problem 2 a
*******
use pisofirme, clear
gen constant = 1
gen s = 1 - cond(danemia==.,1,0)
gen ls_incomepc =log(S_incomepc+1)
gmm (danemia - logistic({xb: dpisofirm S_age S_HHpeople ls_incomepc})) , instruments(dpisofirm S_age S_HHpeople ls_incomepc
seed(123) nodots)
estout using hw3_q2_2a_stata.tex, cells("b ci") style(tex)  replace


*******
*** Problem 2 3c
*******
glm s S_age S_HHpeople ls_incomepc dpisofirme, family(binomial) link(logit)
predict prop_score
gen w_s_age = S_age/prop_score
gen w_s_hhpeople = S_HHpeople/prop_score
gen w_ls_incomepc = ls_incomepc/prop_score
gen w_dpisofirme = dpisofirme/prop_score
gmm (danemia - logistic({xb: S_age S_HHpeople ls_incomepc dpisofirme})), instruments(w_*, noconstant)
estout using hw3_q2_3c_stata.tex, cells("b ci") style(tex)  replace


*******
*** Problem 2 3d
*******
drop if prop_score < 0.1
gmm (danemia - logistic({xb: S_age S_HHpeople ls_incomepc dpisofirme}) ), instruments(w_*, noconstant) vce(bs, r(49)
seed(123) nodots)
estout using hw3_q2_3d_stata.tex, cells("b ci") style(tex)  replace



*******
*** Problem 3 a
*******
clear all
set obs 1000
set seed 123
gen x = runiform()

sum x
local max_x = r(max)
bs max_x_star = r(max) , reps(599) saving(mbs,replace): sum x
use mbs, clear
gen stat = 1000*(`max_x' - max_x_star)
twoway (histogram stat, bin(16) ) (function exp(-x),range(0 8)) ,title("Distribution of Bootstrap Statistic")
```

```
gr export hw3_Q3_1_stata.png ,replace



*******
*** Problem 3 b
*******
clear all
set obs 1000
set seed 123
gen x = runiform()
sum x
local max_x = r(max)
di `max_x'


program pbs, rclass
        args max_x
        drop _all
        set obs 1000
        gen x_pbs = runiform(0,`max_x')
        egen max_x_pbs = max(x_pbs)
        drop if _n>1
end
pbs `max_x'

simulate max_pbs= max_x_pbs, reps(599): pbs `max_x'


gen stat = 1000*(`max_x' - max_pbs)
twoway (histogram stat, bin(16) ) (function exp(-x),range(0 8)), title("Distribution of Parametric Bootstrap Statistic")
gr export hw3_Q3_2_stata.png ,replace
```

R

```
####################################################################
# ECON 675, Assignment 3
# Fall 2018
# University of Michigan
# Latest update: Oct 22, 2018
####################################################################

rm(list=ls(all=TRUE))
library(foreign); library(MASS);
library(boot)
library(data.table)
library(foreach)
library(data.table)
library(Matrix)
library(ggplot2)
library(sandwich)
library(xtable)
library(gmm)


setwd("/Users/erinmarkiewitz/Dropbox/Phd_Coursework/Econ675/hw3")

# load the data
pisofirme <- read.csv("pisofirme.csv", header = TRUE)
complete  <- complete.cases(pisofirme[, 5:27])
pisofirme <- pisofirme[complete, ]
# s_i: non-missing indicator
pisofirme$log_inc <- log(pisofirme$S_incomepc+1)
pisofirme$nmissing <- 1 - pisofirme$dmissing
pisoframe = as.data.frame(pisofirme)


# Get Piso Firme data
pisoframe <- as.data.table(read.csv('pisofirme.csv'))

# Create dependent variable for logistic regression
pisoframe[,nmissing:= 1-dmissing]

# Create income regressor
pisoframe[,log_inc:= log(S_incomepc+1)]

# Create income regressor
pisoframe[,log_inc:= log(S_incomepc+1)]

####################################################################
# Q 1_9 a
####################################################################

#estimate logit model
logit_q1 <- glm(nmissing ~ S_age + S_HHpeople + log_inc,
    family = "binomial", data = pisoframe)

#extract point estimates and calculate standard errors
b.hat   <- as.data.table(logit_q1["coefficients"])
```

```
V.hat   <- vcovHC(logit_q1, type = "HC1")
se.hat  <- as.data.table(sqrt(diag(V.hat)))
V.out <- diag(V.hat)
#compute t-stats and p values
t.stat <- b.hat/se.hat
n = nrow(pisofirme)
d = 4
p = round(2*pt(abs(t.stat[[1]]),df=n-d,lower.tail=FALSE),3)

#compute CI
CI.lower = b.hat - qnorm(0.975)*se.hat
CI.upper = b.hat + qnorm(0.975)*se.hat

results.a  = as.data.frame(cbind(b.hat,V.out,t.stat,p,CI.lower,CI.upper))
colnames(results.a) = c("Coef.","V","t-stat","p-val","CI.lower","CI.upper")
rownames(results.a) = c("Const.", "S_age","S_HHpeople","log_inc")

# Get latex table output
xtable(results.a,digits=5)
print(xtable(results.a, type = "latex"), file = "hw3_q1_9a_r.tex")

###################################################################
# Q 1_9 b
###################################################################


# set up logistic bootstrap
boot.logit <- function(data, i){
  logit  <- glm(nmissing ~ S_age + S_HHpeople + +I(log(S_incomepc+1)),
                data = data[i, ], family = "binomial")
  V      <- vcovHC(logit, type = "HC1")
  se     <- sqrt(diag(V.hat))
  t.boot <- (coef(logit)-coef(logit_q1))/se

  return(t.boot)
}

# run logistic bootstrap
set.seed(123)
boot.out <- boot(data=pisofirme, R=499, statistic = boot.logit, stype = "i")

# back out quantiles of boot t-dist. for CIs
boot.quant <- sapply(1:4, function (i) quantile(boot.out$t[,i], c(0.025, 0.975)))

#CIs
boot.ci.lower = b.hat + t(boot.quant)[,1]*se.hat
boot.ci.upper = b.hat + t(boot.quant)[,2]*se.hat

boot.p = sapply(1:4,function(i) 1/499*sum(boot.out$t[,i]>=t.stat[i]))

# Tabulate bootstrap results
results.b  = as.data.frame(cbind(b.hat,boot.ci.lower,boot.ci.upper,boot.p))
colnames(results.b) = c("Coef.","CI.lower","CI.upper","p-val")
rownames(results.b) = c("Const.", "S_age","S_HHpeople","log_inc")

# Get latex table output
xtable(results.b,digits=4)
print(xtable(results.b, type = "latex"), file = "hw3_q1_9b_r.tex")

###################################################################
# Q 1_9 C
###################################################################

# subset data
X  = pisoframe[, c("S_age","S_HHpeople","log_inc")]
X$const = 1
setcolorder(X,c("const","S_age","S_HHpeople","log_inc"))
b.hat = coef(logit_q1)

# Construct link function
mu = function(u){(1+exp(-u))^(-1)}

# Construct vector of x_i'*beta.hats
XB = as.matrix(X)%*%b.hat
# Compute predicted probabilities
mu.hat = mu(XB)
X[,mu.hat:=mu.hat]

#Make plot
plot(density(mu.hat,kernel="e", adjust = 5, bw="ucv",na.rm=TRUE),main="Kernel Density Estimate")
dev.copy(png,'hw3_q1_9c_r.png')
dev.off()


###################################################################
# ECON 675, Assignment 3
# Fall 2018
# University of Michigan
# Latest update: Oct 22, 2018
###################################################################
```

```
rm(list=ls(all=TRUE))
library(foreign); library(MASS);
library(boot)
library(data.table)
library(foreach)
library(data.table)
library(Matrix)
library(ggplot2)
library(sandwich)
library(xtable)
library(gmm)


setwd("/Users/erinmarkiewitz/Dropbox/Phd_Coursework/Econ675/hw3")

# load the data
pisofirme <- read.csv("pisofirme.csv", header = TRUE)
complete  <- complete.cases(pisofirme[, 5:27])
pisofirme <- pisofirme[complete, ]
# s_i: non-missing indicator
pisofirme$log_inc <- log(pisofirme$S_incomepc+1)
pisofirme$nmissing <- 1 - pisofirme$dmissing
pisoframe = as.data.frame(pisofirme)


# Get Piso Firme data
pisoframe <- as.data.table(read.csv('pisofirme.csv'))

# Create dependent variable for logistic regression
pisoframe[,nmissing:= 1-dmissing]

# Create income regressor
pisoframe[,log_inc:= log(S_incomepc+1)]

# Create income regressor
pisoframe[,log_inc:= log(S_incomepc+1)]




##################################################################
# Q 2.2 MCAR
##################################################################
# GMM moment condition: logistic
g_logistic <- function(theta, data) {
  a <- (data$danemia - plogis(theta[1]*data$dpisofirme + theta[2]*data$S_age + theta[3]*data$S_HHpeople + theta[4]*log(1+da
    data$dpisofirme
  b <- (data$danemia - plogis(theta[1]*data$dpisofirme + theta[2]*data$S_age + theta[3]*data$S_HHpeople + theta[4]*log(1+da
    data$S_age
  c <- (data$danemia - plogis(theta[1]*data$dpisofirme + theta[2]*data$S_age + theta[3]*data$S_HHpeople + theta[4]*log(1+da
    data$S_HHpeople
  d <- (data$danemia - plogis(theta[1]*data$dpisofirme + theta[2]*data$S_age + theta[3]*data$S_HHpeople + theta[4]*log(1+da
    log(1+data$S_incomepc)
  cbind(a, b, c, d)
}

# logistic bootstrap
boot.T_logistic <- function(boot.data, ind) {
  gmm(g_logistic, boot.data[ind, ], t0=c(0,0,0,0), wmatrix="ident", vcov="iid")$coef
}
ptm <- proc.time()
set.seed(123)
temp <- boot(data=pisofirme[pisofirme$nmissing==1, ], R=499, statistic = boot.T_logistic, stype = "i")
proc.time() - ptm
table3 <- matrix(NA, ncol=6, nrow=4)
for (i in 1:4) {
  table3[i, 1] <- temp$t0[i]
  table3[i, 2] <- sd(temp$t[, i])
  table3[i, 3] <- table3[i, 1] / table3[i, 2]
  table3[i, 4] <- 2 * max( mean(temp$t[, i]-temp$t0[i]>=abs(temp$t0[i])), mean(temp$t[, i]-temp$t0[i]<=-1*abs(temp$t0[i]))
  table3[i, 5] <- 2 * temp$t0[i] - quantile(temp$t[, i], 0.975)
  table3[i, 6] <- 2 * temp$t0[i] - quantile(temp$t[, i], 0.025)
}

rownames(table3)=c("dpisofirme", "S_age","S_HHpeople","log_inc")
colnames(table3)=c("Estimate", "Std.Error", "t", "p-value", "CI.lower","CI.upper")

xtable(table3,digits=3)
print(xtable(table3, type = "latex"), file = "hw3_q2_2_r.tex")


##################################################################
# Q 2.3 MAR
##################################################################
# GMM moment condition
g_MAR <- function(theta, data) {
  data <- data[data$nmissing==1, ]
  a <- (data$danemia - plogis(theta[1]*data$dpisofirme + theta[2]*data$S_age + theta[3]*data$S_HHpeople + theta[4]*log(1+da
    data$dpisofirme * data$weights
  b <- (data$danemia - plogis(theta[1]*data$dpisofirme + theta[2]*data$S_age + theta[3]*data$S_HHpeople + theta[4]*log(1+da
    data$S_age * data$weights
  c <- (data$danemia - plogis(theta[1]*data$dpisofirme + theta[2]*data$S_age + theta[3]*data$S_HHpeople + theta[4]*log(1+da
```

```
    data$S_HHpeople * data$weights
  d <- (data$danemia - plogis(theta[1]*data$dpisofirme + theta[2]*data$S_age + theta[3]*data$S_HHpeople + theta[4]*log(1+da
    log(1+data$S_incomepc) * data$weights
  cbind(a, b, c, d)
}

# logistic bootstrap
boot.T_MAR <- function(boot.data, ind) {
  data.temp <- boot.data[ind, ]
  fitted <- glm(nmissing ~ dpisofirme + S_age + S_HHpeople +I(log(S_incomepc+1)) - 1,
                data = data.temp,
                family = binomial(link = "logit"))$fitted
  data.temp$weights <- 1 / fitted
  gmm(g_MAR, data.temp, t0=c(0,0,0,0), wmatrix="ident", vcov="iid")$coef
}

ptm <- proc.time()
set.seed(123)
temp <- boot(data=pisofirme, R=499, statistic = boot.T_MAR, stype = "i")
proc.time() - ptm
table5 <- matrix(NA, ncol=6, nrow=4)
for (i in 1:4) {
  table5[i, 1] <- temp$t0[i]
  table5[i, 2] <- sd(temp$t[, i])
  table5[i, 3] <- table5[i, 1] / table5[i, 2]
  table5[i, 4] <- 2 * max( mean(temp$t[, i]-temp$t0[i]>=abs(temp$t0[i])), mean(temp$t[, i]-temp$t0[i]<=-1*abs(temp$t0[i]))
  table5[i, 5] <- 2 * temp$t0[i] - quantile(temp$t[, i], 0.975)
  table5[i, 6] <- 2 * temp$t0[i] - quantile(temp$t[, i], 0.025)
}
filename <- paste("logistic_boot_MAR.txt")


rownames(table5)=c("dpisofirme", "S_age","S_HHpeople","log_inc")
colnames(table5)=c("Estimate", "Std.Error", "t", "p-value", "CI.lower","CI.upper")

xtable(table5,digits=3)
print(xtable(table5, type = "latex"), file = "hw3_q2_3c_r.tex")

# GMM moment condition with trimming
g_MAR2 <- function(theta, data) {
  data <- data[data$nmissing==1 & data$weights<=1/0.1, ]
  a <- (data$danemia - plogis(theta[1]*data$dpisofirme + theta[2]*data$S_age + theta[3]*data$S_HHpeople + theta[4]*log(1+da
    data$dpisofirme * data$weights
  b <- (data$danemia - plogis(theta[1]*data$dpisofirme + theta[2]*data$S_age + theta[3]*data$S_HHpeople + theta[4]*log(1+da
    data$S_age * data$weights
  c <- (data$danemia - plogis(theta[1]*data$dpisofirme + theta[2]*data$S_age + theta[3]*data$S_HHpeople + theta[4]*log(1+da
    data$S_HHpeople * data$weights
  d <- (data$danemia - plogis(theta[1]*data$dpisofirme + theta[2]*data$S_age + theta[3]*data$S_HHpeople + theta[4]*log(1+da
    log(1+data$S_incomepc) * data$weights
  cbind(a, b, c, d)
}

# logistic bootstrap
boot.T_MAR2 <- function(boot.data, ind) {
  data.temp <- boot.data[ind, ]
  fitted <- glm(nmissing ~ dpisofirme + S_age + S_HHpeople +I(log(S_incomepc+1)) - 1,
                data = data.temp,
                family = binomial(link = "logit"))$fitted
  data.temp$weights <- 1 / fitted
  gmm(g_MAR2, data.temp, t0=c(0,0,0,0), wmatrix="ident", vcov="iid")$coef
}

ptm <- proc.time()
set.seed(123)
temp <- boot(data=pisofirme, R=499, statistic = boot.T_MAR2, stype = "i")
proc.time() - ptm
table6 <- matrix(NA, ncol=6, nrow=4)
for (i in 1:4) {
  table6[i, 1] <- temp$t0[i]
  table6[i, 2] <- sd(temp$t[, i])
  table6[i, 3] <- table6[i, 1] / table6[i, 2]
  table6[i, 4] <- 2 * max( mean(temp$t[, i]-temp$t0[i]>=abs(temp$t0[i])), mean(temp$t[, i]-temp$t0[i]<=-1*abs(temp$t0[i]))
  table6[i, 5] <- 2 * temp$t0[i] - quantile(temp$t[, i], 0.975)
  table6[i, 6] <- 2 * temp$t0[i] - quantile(temp$t[, i], 0.025)
}

rownames(table5)=c("dpisofirme", "S_age","S_HHpeople","log_inc")
colnames(table5)=c("Estimate", "Std.Error", "t", "p-value", "CI.lower","CI.upper")

xtable(table6,digits=3)
print(xtable(table6, type = "latex"), file = "hw3_q2_3d_r.tex")


################################################################
# ECON 675, Assignment 3
# Fall 2018
# University of Michigan
# Latest update: Oct 22, 2018
################################################################


rm(list=ls(all=TRUE))
```

```
library(foreign); library(MASS);
library(boot)
library(data.table)
library(foreach)
library(data.table)
library(Matrix)
library(ggplot2)
library(sandwich)
library(xtable)
library(gmm)

######################################################################
# Q3 1
######################################################################
set.seed(123)
# Set up Enviorment
N = 1000
X = runif(N,0,1)
x.max = max(X)

# Write function for bootrap statistic
boot.stat = function(data, i){
  N*(x.max -max(data[i]))
}

# Run bootsrap with 599 replications
boot.results = boot(data = X, R = 599, statistic = boot.stat)

# Make frequency plot
h          = hist(boot.results$t,plot=FALSE)
h$density = h$counts/sum(h$counts)
plot(h,freq=FALSE,main="Distribution of Bootstrap Statistic",xlab="Bootstrap statistic")
dev.copy(png,'hw3_q3_2_r.png')
dev.off()

######################################################################
# Q3: 2
######################################################################

# Generate parametric bootstrap samples
X.boot = replicate(599,runif(N,0,x.max))

# Compute maximums for each replications
x.max.boot = sapply(1:599,function(i) max(X.boot[,i]))

# Compute bootstrap statistic
t.boot      = N*(x.max -x.max.boot)

x.quant = range(c(0, 1, 100))
x.exp = dexp(x.quant, rate = 1, log = FALSE)

# Make frequency plot
h2          = hist(t.boot,plot=FALSE)
h2$density = h2$counts/sum(h2$counts)
plot(h2,freq=FALSE,main="Distribution of Parametric Bootstrap Statistic",xlab="Parametric bootstrap statistic",ylim=c(0,0.4
dev.copy(png,'hw3_q3_3_r.png')
dev.off()
```