1. (a) Let \( \hat{f} \) denote \( \hat{f}(x) \)

\[
E[(y - \hat{f})^2] = E[(\hat{f} - y)^2] = E[(\hat{f} - f - \varepsilon)^2] \text{ as } y = f + \varepsilon \\
= E[(\hat{f} - f - \varepsilon)^2] + E[\varepsilon^2] \text{ as } \varepsilon \perp X \text{ and } E[\varepsilon] = 0 \\
= E[(\hat{f} - E[\hat{f}]) + (E[\hat{f}] - f)]^2 + E[\varepsilon^2] \\
= E[(\hat{f} - E[\hat{f}])^2] + E[\varepsilon^2] \text{ as cross term } = 0 \\
= Var[\hat{f}] + \{Bias(\hat{f})\}^2 + Var(\varepsilon).
\]

(b) 10-fold cross-validation. Randomly divide data into 10 folds of approximately equal size. In turn drop one fold, estimated on the other nine folds, and then predict and compute MSE in the dropped fold. Then \( CV_{(10)} = \frac{1}{10} \sum_{k=1}^{K} MSE_{(10)} \). The model is OLS regression of \( y \) on \( x, x^2, ..., x^p \). For \( p = 1, 2, ..., \) compute \( CV^{(p)}_{(10)} \) and choose degree \( p^* \) for which \( CV^{(p)}_{(10)} \) is lowest.

(c) Lasso estimator \( \hat{\beta}_\lambda \) of \( \beta \) minimizes \( \sum_{i=1}^{n}(y_i - x_i^T \beta)^2 + \lambda \sum_{j=1}^{p} |\beta_j| =RSS + \lambda ||\beta||_1 \), where \( \lambda \geq 0 \) is a tuning parameter and \( ||\beta||_1 = \sum_{j=1}^{p} |\beta_j| \) is L1 norm. Equivalently the lasso estimator minimizes \( \sum_{i=1}^{n}(y_i - x_i^T \beta)^2 \) subject to \( \sum_{j=1}^{p} |\beta_j| \leq s \).

(d) Splines are one example. For example, cubic spline estimate by OLS \( f(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4(x-c_1)^3_+ + \cdots + \beta_{(3+K)}(x-c_K)^3_+ \), where \( (x-c)_+ = (x-c) \text{ if } (x-c) > 0 \) and equals 0 otherwise.

Or smoothing spline minimize \( \sum_{i=1}^{n}(y_i - g(x_i))^2 + \lambda \int_{a}^{b} g''(t) dt \) where \( a \leq x_i \leq b \). Local polynomial and neural networks also possible.

(e) First cut. Choose \( y_* \) for which \( \sum_{i \in R_1}(y_i - \bar{y}_{R_1})^2 + \sum_{i \in R_2}(y_i - \bar{y}_{R_2})^2 \) is smallest, where \( R_1 = \{y_i : y_i \leq y^* \} \) and \( R_2 = \{y_i : y_i > y^* \} \). Second cut now do splits on \( R_1 \) and \( R_2 \) to minimize \( \sum_{j=1}^{3} \sum_{i \in R_j}(y_i - \bar{y}_{R_j})^2 \).

(f) Logistic regression specifies a model for \( \Pr[y | X = x] \). i.e. conditional on \( x \), linear discriminant analysis specifies a model for \( f(X | y) \) and for \( \Pr[y] \). i.e. Joint \( X \) and \( y \). It then backs out \( \Pr[y | X = x] \) using Bayes rule.

(g) Principal components. Several ways to explain. Simplest is that the first principal component has the largest sample variance among all normalized linear combinations of the columns of \( X \). The second principal component has the largest variance subject to being orthogonal to the first, and so on.

K-Means splits into \( K \) distinct clusters where within cluster variation (measured using Euclidean distance, for example) is minimized.

2. (a) \( L(y | \theta) = \prod_{i=1}^{n} e^{-\theta y_i / y_i!} = e^{-\theta \bar{y} / (\prod_{i=1}^{n} y_i!)} \) and \( \pi(\theta) = 1 \).

Posterior density \( \pi(\theta | y) = e^{-\theta \bar{y} / (\prod_{i=1}^{n} y_i!)} \times 1 \propto e^{-n \theta \bar{y}} \).

(b) \( \pi(\theta | y) \propto e^{-n \theta (\bar{y} + 1) - 1} \) which is gamma density with \( a = (n \bar{y} + 1) \) and \( b = n \) so posterior mean equals \( a/b = (n \bar{y} + 1)/n = \bar{y} + (1/n) \).

(c) Use Gibbs sampling. At \( s^{th} \) round draw \( \theta_1^{(s)} \) from \( \pi(\theta_1 | \theta_2^{(s-1)}) \) and then \( \theta_2^{(s)} \) from \( \pi(\theta_2 | \theta_1^{(s-1)}) \).

(d) For latent variable models where if the latent variable was fully observed analysis would be straightforward. examples are logit and probit and, for missing data problems, multivariate normal.
(e) Use Metropolis-Hastings when it is difficult to make draws directly from the posterior density. Instead make draws from a candidate density and keep the draw given the MH rule.

(f) In Stata generate \( \text{expb} = \exp(b) \) and then \( \text{sum expb} \).
In Stata \( \text{centile expb, centile(2.5,9.5)} \). Since \( \beta > 0 \) always this will just yield \( (e^{1706708}, e^{1.019916}) \).

(g) The autocorrelations are very slow to die out; the first 100 draws after burnin have only one new value accepted; the trace has some big movements; the density is bimodal (though the first and second halves are similar).
I would be concerned that the chain has not converged.

3. (a) \( \hat{\beta} = \left( \sum_{g=1}^{G} \sum_{i=1}^{N_g} x_{ig} x'_{ig} \right)^{-1} \left( \sum_{g=1}^{G} \sum_{i=1}^{N_g} x_{ig} y_{ig} \right) \)
\[ = \left( \sum_{g=1}^{G} x_{ig} x'_{ig} \times \sum_{i=1}^{N_g} \right)^{-1} \left( \sum_{g=1}^{G} x_{ig} \times \sum_{i=1}^{N_g} y_{ig} \right) \]
\[ = \left( \sum_{g=1}^{G} x_{ig} x'_{ig} \times N_g \right)^{-1} \left( \sum_{g=1}^{G} x_{ig} \times N_g \bar{y}_g \right) \text{ where } \bar{y}_g = \frac{1}{N_g} \sum_{i=1}^{N_g} y_{ig} \]
\[ = \left( \sum_{g=1}^{G} x_{ig} x'_{ig} \right)^{-1} \left( \sum_{g=1}^{G} x_{ig} \bar{y}_g \right) \text{ which is OLS regression of } \bar{y}_g \text{ on } x_g. \]

(b) \( \text{Var}(\bar{y}_g) = \text{Var}\left( \frac{1}{N_g} \sum_{i=1}^{N_g} u_{ig} \right) = \frac{1}{N_g^2} \text{Var}\left( \sum_{i=1}^{N_g} u_{ig} \right) \)
\[ = \frac{1}{N_g^2} \left( \sum_{i=1}^{N_g} \text{Var}(u_{ig}) + \sum_{i=1}^{N_g} \sum_{j=1: j \neq i}^{N_g} \text{Cov}(u_{ig}, u_{ij}) \right) = \frac{1}{N_g^2} (N_g \sigma_u^2 + N_g(N_g - 1) \rho \sigma_u^2) = \frac{\sigma_u^2}{N_g} (1 + (N_g - 1) \rho). \]

(c) \( \text{Var}\left( \sum_{g=1}^{G} x_{ig} \bar{y}_g \right) = \sum_{g=1}^{G} \text{Var}(x_{ig} \bar{y}_g) = \sum_{g=1}^{G} \text{Var}(\bar{y}_g) x_{ig} x'_{ig} \)
\[ = \sum_{g=1}^{G} \frac{\sigma_u^2}{N_g} (1 + (N_g - 1) \rho) x_{ig} x'_{ig} = \frac{\sigma_u^2}{N_g} (1 + (N_g - 1) \rho) \sum_{g=1}^{G} x_{ig} x'_{ig}. \]
So \( \text{Var}(\hat{\beta}) = \left( \sum_{g=1}^{G} x_{ig} x'_{ig} \right)^{-1} \text{Var}\left( \sum_{g=1}^{G} x_{ig} \bar{y}_g \right) \left( \sum_{g=1}^{G} x_{ig} x'_{ig} \right)^{-1} = \frac{\sigma_u^2}{N_g} (1 + (N_g - 1) \rho) \left( \sum_{g=1}^{G} x_{ig} x'_{ig} \right)^{-1}. \]

(d) The default uses \( \text{Var}(\bar{y}_g) = \frac{\sigma_u^2}{N} \) so \( \text{Var}(\hat{\beta}) = \frac{\sigma_u^2}{N^2} \left( \sum_{g=1}^{G} x_{ig} x'_{ig} \right)^{-1}. \)
The true variance is \((1 + (N_g - 1) \rho)\) times this.

(e) No. The key is that \( \frac{1}{G} \sum_{g=1}^{G} X'_{tg} \tilde{u}_g X_g - \frac{1}{G} \sum_{g=1}^{G} X'_{tg} u_i u'_{tg} X_g \rightarrow 0. \)

(f) \( \text{Var}(\tilde{\theta}) = \left( \sum_{g=1}^{G} \sum_{i} \frac{\partial \text{m}(y_{ig}, x_{ig}, \theta)}{\partial \theta} \right)^{-1} \left( \sum_{g=1}^{G} \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} \text{E}[\text{m}(y_{ig}, x_{ij}, \theta) \text{m}(y_{ig}, x_{ij}, \theta)'] \right) \left( \sum_{g=1}^{G} \sum_{i} \frac{\partial \text{m}(y_{ig}, x_{ig}, \theta)}{\partial \theta} \right) \)

(g) Various methods. Cluster pairs bootstrap resamples the clusters with replacement \( B \) times and computes \( B \) estimates \( \hat{\theta} \). Then estimated variance is \( \frac{1}{B} \sum_{b=1}^{B} (\hat{\theta}^{(b)} - \hat{\theta}) (\hat{\theta}^{(b)} - \hat{\theta})' \) where \( \hat{\theta} = \frac{1}{B} \sum_{b=1}^{B} \hat{\theta} \). Take square root of diagonal entries to get standard errors. This does no better with small clusters than the usual cluster-robust estimate.

(h) Denote the two ways of clustering \( G \) and \( H \). OLS with cluster one-way on \( G \) gives variance matrix \( \hat{V}_G \). Similar on \( H \) gives \( \hat{V}_H \). Similar on \( G \cap H \) gives \( \hat{V}_{G \cap H} \). Then \( \hat{V}_{2way} = \hat{V}_G + \hat{V}_H - \hat{V}_{G \cap H} \).