SKETCH of the SOLUTIONS
STK-IN4300/STK-IN9300 - 2019

Problem 1

a Ridge regression, because it performs better in the case of many variables with small effect (which can be seen from $\beta \sim N(0, 2)$) and if there is a strong correlation among the variables ($\rho = 0.9$).

b Without knowing that there are many variables with small effect (i.e., we miss the information on the βs), it is safer to use lasso “betting on sparsity”: if there are only a few variables with a large effect, it may strongly outperform ridge; if the situation is similar to the one at point (a), it will not perform much worse than ridge regression.

NOTE: if there was good reasoning behind the choice of ridge regression, the answer was considered correct. Example of good reasoning: “**Despite** the fact that I do not know the effect of the variables, which would let me choose lasso, I prefer to use ridge because I want my model to handle the correlation in a better way, and, anyway, part of the effect of a few potential strong variables is shared with all the variables due to correlation”.

c Yes, it would allow having a sparse model due to the L_1 penalty, with a better handling of the correlation among variables thanks to the L_2 penalty.

Problem 2

a The idea is to start from a parametric estimate of the density, $f_0(x, \theta)$, and later correct it with a non-parametric part ($\sum_{i=1}^{N} \frac{K_\lambda(x_i - x)}{f_0(x_i, \hat{\theta})}$). So one could have a first good global approximation, and let the non-parametric part focus on the discepancies between the parametric part and the true density function.

b In this case, one may want to use an exponential distribution, that solves the boundary problem: being equal to 0 for impossible values (there cannot be a negative quantity of wine drunk by a person in one year), it prevents the non-parametric part to give positive density to areas outside the support.

Exercise 3

a The procedure is wrong because it uses the test data in the model construction procedure, and the error computed in the second point is underestimated. The cross-validation procedure to find the best value of the tuning parameter, indeed, must use only the training data. The correct procedure is to use a nested cross-validation procedure: for example, use LOOCV and, in each iteration, perform the K-fold cross-validation on the $n-1$ observations currently used as a training set to find the best tuning parameter.

b The larger the number of folds, smaller the bias, because the training sets will be larger, more similar to the whole training set; and larger the variance, because the training sets will be very similar to each other, so the estimate will be very sample-specific: with a different dataset, one could obtain a very different estimate of the error.

NOTE: in this point was very important to show to have understood that the variance refers to the whole cross-validation estimator and not to the fold-specific estimator.

Exercise 4

a We expect the orange line to go under the green line, i.e., the result of consensus being worse than the individual guess. It shows that when the estimators computed in each repetition of bagging are worse than guessing at random, in average bagging actually performs worse than the single estimator, as it "makes stronger" a bad solution.

b Since 35 out of 50 members vote randomly (so they are correct with probability 1/4) and the remaining 15 with probability p,

$$EC = 10 \times \left(\frac{35}{50} \cdot \frac{1}{4} + \frac{15}{50} \cdot p\right)$$

where EC is the expeced number of correct answers in the 10 categories. With $p = 0$, $EC = 10 \times 7/40 = 7/4$.

c Since $Var\left[\frac{1}{B}\sum_{b=1}^{B} X_b\right] = E\left[(\frac{1}{B}\sum_{b=1}^{B} X_b)^2\right] - \left(E[(\frac{1}{B}\sum_{b=1}^{B} X_b]\right)^2$, with the last term simply equal to μ^2, we should focus on the first term on the right hand side,

$$
\begin{aligned}
E\left[(\frac{1}{B}\sum_{b=1}^{B} X_b)^2\right] &= \frac{1}{B^2} E\left[\sum_{b=1}^{B} X_b^2 + 2\sum_{b\neq c} X_b X_c\right] \\
&= \frac{1}{B^2} E\left(\sum_{b=1}^{B} E[X_b^2] + 2\sum_{b>c} E[X_b X_c]\right) \\
&= \frac{1}{B^2} B(\mu^2+\sigma^2) + \frac{2}{B^2}\binom{B}{2} E[X_b X_c] \\
&= \frac{1}{B}(\mu^2+\sigma^2) + \frac{2}{B^2}\frac{B(B-1)}{2}(\rho\sigma^2+\mu)
\end{aligned}
$$

because, given that $E[X_b] = E[X_c] = \mu$,

$$
\begin{aligned}
\rho &= \frac{E[(X_b-\mu)(X_c-\mu)]}{\sigma^2} \\
&= \frac{E[X_b X_c] - \mu E[X_c] - \mu E[X_b] + \mu^2}{\sigma^2} \\
&= \frac{E[X_b X_c] - \mu^2}{\sigma^2}
\end{aligned}
$$

so that $E[X_b X_c] = \rho\sigma^2 + \mu^2$.

Substituting in the first espression,

$$
\begin{aligned}
Var\left[\frac{1}{B}\sum_{b=1}^{B} X_b\right] &= E\left[(\frac{1}{B}\sum_{b=1}^{B} X_b)^2\right] - \left(E[(\frac{1}{B}\sum_{b=1}^{B} X_b]\right)^2 \\
&= \frac{1}{B}(\mu^2+\sigma^2) + \frac{2}{B^2}\frac{B(B-1)}{2}(\rho\sigma^2+\mu^2) - \mu^2 \\
&= \frac{1}{B}(\mu^2+\sigma^2) + \frac{B-1}{B}(\rho\sigma^2+\mu^2) - \mu^2 \\
&= \frac{\mu^2+\sigma^2+B\rho\sigma^2-\rho\sigma^2+B\mu^2-\mu^2-B\mu^2}{B} \\
&= \frac{\sigma^2+B\rho\sigma^2-\rho\sigma^2}{B} \\
&= \frac{\sigma^2}{B} + \rho\sigma^2 - \frac{\rho\sigma^2}{B} \\
&= \rho\sigma^2 + \frac{(1-\rho)}{B}\sigma^2
\end{aligned}
$$

Exercise 4

a If the boosting step size is too small, it would take too many iterations to reach the best estimate in terms of prediction error. We see that in the first plot, in which we do not find the minimum within the considered number of boosting iterations. On the other hand, if the boosting step size is too large, we do not have enough shrinkage and we risk to overfit already at the beginning, as we can see in the fourth figure. When the boosting step size has the right magnitude, we see the typical behaviour of the prediction error due to the combination of bias and variance. By increasing the number of boosting steps, initially we strongly decrease the bias, and the prediction error decreases despite the increase of the variance. At a certain point, however, the bias reduction gets small, and the increase of the variance will dominate. As a consequence, the prediction error increases (we are overfitting). In this case, one may want to choose $\nu = 0.1$ and 8 iterations: one could get a similar result with $\nu = 0.01$ and 55 boosting iterations, but it would need more time without any improvement in terms of performance.

NOTE: those who chose $\nu = 0.01$ and number iteration $= 55$ providing a good reason (e.g., the curve is more smooth therefore any small error in the choice of the number of steps will affect way less the algorithm) got full points.

b In the componentwise version of boosting one dimension is updated at each iteration. Advantages includes the possibility to implement boosting when we have more variables than observations (high-dimensional data), automatic variable selection (the irrelevant dimensions are never selected as the best direction to get improved and therefore never included in the model), and the possibility to use different base-learners for each dimension (e.g., linear effects and splines).