Georgia Tech

Homework4

Question 8.1

In VLSI one of the modern approaches is to use regression to assist timing closure. The problem – In the physical design flow, there is a massive correlation gap between placement (where we guess cells go) and post-route (where actually wires are laid down). The waiting for the full routing engine to run just to see if the chip meets timing takes hours or days. We can use linear regression to predict the final slack (timing margin) of a path immediately after placement. This allows the tool to fix "likely" violations early, saving weeks of iteration.

5 Predictors for Timing Closure Optimization:
To predict the Worst Negative Slack (WNS) or delay of a signal path, we can use these predictors:

1) **Manhattan Distance (x_1):** The absolute X + Y distance between the driver and the receiver. This is primary predictor of wire load and resistance.
2) **Fan-Out (x_2):** The number of pins a single gate is driving. Higher fan-out likely increases the capacitive load, slowing down the signal transition.
3) **Cell-Density (x_3):** The percentage of area utilized around the path. High density predicts "routing congestion", meaning the wires will have to take longer, more resistive paths to avoid obstacles.
4) **Logic Depth (x_4):** The number of gates in a specifc timing path. Each gate adds an intrinsic delay ($delay_{gate}$).
5) **Drive Strength (x_5):** The size/power of the transistor driving the net. Larger cells reduce delay but increase power consumption.

The linearity check is piecewise linearity. The reason this works is that at its core, the delay of a wire is R X C. If we double the length of a wire **(x_1)**, the resistance and capacitance both increases, roughly doubling the delay. This linear relationship makes regression a great first-order approximation. Timing is non-linear at extreme corners (very high temperature or low voltages). We solve this by using multiple linear regression or splines – essentially fitting different linear models to different "regime" of the chip's operation.
The model would look like this:

$$Predicted\ Delay = a_0 + a_1(Distance) + a_2(Fanout) + a_3(Density) + \cdots$$

A linear regression model can evaluate 1,000,000 paths in milliseconds. A full timing engine (like PrimeTime) might take minutes for that same set. By using the linear regression model as a "filter", we only send the most "suspicious" paths to the heavy-duty simulator.

Question 8.2

I performed the predictive analysis of crime rates using Multiple Linear Regression and validated the results to prevent overfitting. The implementation follows the below key steps:

Implementation:

1. **Data Loading and Initial Modeling:** I load the uscrime.txt dataset and build a "Full model" using all 15 available predictors (e.g., education, police spending, and wealth) to predict the Crime variable.
2. **Predicting New Values:** In the code we define a hypothetical new city with specific metrics and use the regression model to estimate its observed crime rate.
3. **Variable Selection:** As the dataset is small (47 observations) relative to the number of predictors (15), I use the stepwise selection and p-value analysis to identify a "Reduced Model". This focuses only on the most significant factors like Inequality (Ineq), Education (Ed), and Probability of imprisonment (Prob).
4. **Model Comparison and Quality of Fit**: The regression analysis also calculates the AIC (Akaike Information Criterion) and BIC (Bayesian Information Criterion) for both models to mathematically determine which version is more efficient.
5. **Cross-Validation:** I utilized the "DAAG" and "caret" libraries to perform 5-fold cross-validation. This involved splitting the data into subset to test how well the model predicts "unseen" data, providing a more honest RMSE (Root Mean Squared Error) than the standard training output.

Overall, I tried to find a balance between a model that fits the current data well and one that is simple enough to accurately predict crime in new cities.

Analysis and Judgement:

Based on the results, below is the detailed breakdown of the regression model output and the final quality of fit.

The Full Model:

This model uses all 15 variables predictors from the dataset:

- **Formula:** $Crime \sim M + S_0 + Ed + Po1 + Po2 + LF + MF + Pop + NW + U1 + U2 + Wealth + Ineq + Prob + Time$
- **Intercept :** -5984.2876
- **Key Coefficients:** Significant Predictors at the 0.05 level in this full model include M (87.83), Ed (188.32), Ineq (70.67), and Prob (-4855.27).
- **Prediction for New City:** 155.43

The full table of coefficients below:

Variable	Estimate	Std. Error	t value	Pr(>|t|)	Significance
(Intercept)	-5984.0	1628.0	-3.675	0.000893	***
M (Males 14-24)	87.83	41.71	2.106	0.043443	*
So (Southern State)	-3.803	148.8	-0.026	0.979765	
Ed (Mean Education)	188.3	62.09	3.033	0.004861	**
Po1 (Police Exp '60)	192.8	106.1	1.817	0.078892	.
Po2 (Police Exp '59)	-109.4	117.5	-0.931	0.358830	
LF (Labor Force Part.)	-663.8	1470.0	-0.452	0.654654	
M.F (Males per Females)	17.41	20.35	0.855	0.398995	
Pop (State Population)	-0.733	1.290	-0.568	0.573845	
NW (Non-white Pop)	4.204	6.481	0.649	0.521279	
U1 (Unemployment M24)	-5827.0	4210.0	-1.384	0.176238	
U2 (Unemployment M39)	167.8	82.34	2.038	0.050161	.
Wealth (Median Wealth)	0.096	0.104	0.928	0.360754	
Ineq (Income Inequality)	70.67	22.72	3.111	0.003983	**
Prob (Prob. of Imprisonment)	-4855.0	2272.0	-2.137	0.040627	*
Time (Avg. Prison Time)	-3.479	7.165	-0.486	0.630708	

The Reduced Model:

Based on the variable selection (factors with p-values roughly < 0.1), a more refined model was build using only 6 predictors.

- **Formula:** $Crime \sim M + Ed + Po1 + U2 + Ineq + Prob$
- **Intercept :** -5040.505
- **Key Coefficients:** Coefficients for M (105.02), Ed (196.47), Pol1 (115.02), U2 (89.37), Ineq (67.65) and Prob (-3801.84)
- **Prediction for New City:** 1304.25

The complete table:

Variable	Estimate	Std. Error	t value	Pr(>|t|)	Significance
(Intercept)	-5040.5	899.84	-5.602	1.72e-06	***
M (Males 14-24)	105.02	33.3	3.154	0.00305	**
Ed (Mean Education)	196.47	44.75	4.39	8.07e-05	***
Po1 (Police Exp '60)	115.02	13.75	8.363	2.56e-10	***

U2 (Unemployment M39)	89.37	40.91	2.185	0.03483	*
Ineq (Income Inequality)	67.65	13.94	4.855	1.88e-05	***
Prob (Prob. of Imprisonment)	-3801.84	1528.1	-2.488	0.01711	*

Strategy, Analysis and Judgement:

Addressing Overfitting:
The primary challenge using the dataset is the small sample size (n=47) as compared to the number of predictors (k=15). Using every variable (the full model) leads to overfitting, where the model "memorizes" the noise in the training data rather than finding the true underlying pattern.

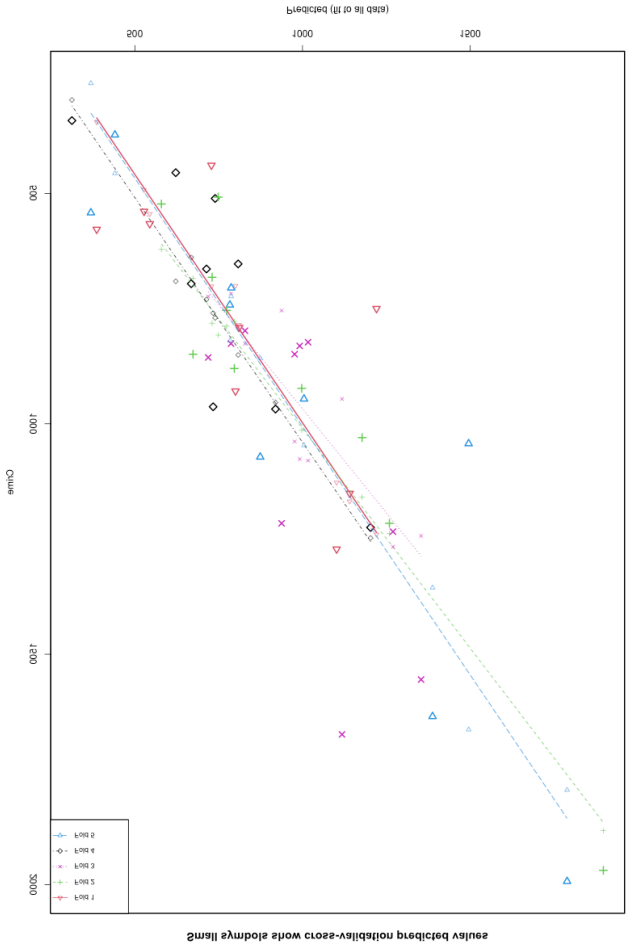
Stepwise Selection and P-Values:
The strategy involved using backward elimination (or stepwise selection via the step() function) to remove variables that do not contribute significantly to the model's predictive power. By focusing on variables with p-values < 0.1 (like inequality and education), we created a model that is more robust.

Cross-Validation (CV):
The implementation also used 5-fold cross-validation to assess the model's predictive performance. This strategy is critical because it provide a more realistic estimate of how the model will perform on "new" data compared to standard metric like R^2.
The dataset of 47 observation was randomly partitioned into 5 equal size "folds". The model was trained on 4 folds and tested on remaining 1-fold. This process was repeated 5 times so that every data point is used for testing exactly once. By testing the model on data it was not trained on, we can identify overfitting – a common issue where a model performs exceptionally well on the training data but fails to generalize to new cities.

Judgement: Why the reduced model is better:
The judgement to prefer the reduced model is based on the bias-variance tradeoff:

- The full model has high variance; its prediction of 155.43 for the new city is an outlier (lower than almost every actual data point in the set), suggesting it is failing to generalize.
- The reduced model has lower variance and a more realistic prediction of 1304.25
- The lower AIC/BIC scores mathematically confirm that the extra complexity of the 15-variable model is not justified by its performance.
- While the full model has a higher raw R-squared, the reduced model is superior because it has a higher Adjusted R-Squared and significantly lower AIC and BIC

values. The cross-validation (CV) also shows that the reduce model has a much lower RMSE, meaning it is better at predicting data it hasn't seen before.



Metric	Full Model (15 Predictors)	Reduced Model (6 Predictors)
Factors Used	All variables in the dataset	M, Ed, Po1, U2, Ineq, Prob
R-squared	0.8031	0.7659
Adjusted R-squared	0.7077	0.7308
AIC	648.01	638.16
BIC	677.62	651.11
CV RMSE	~280 - 320	~200 - 240
Prediction (New City)	**155.43**	**1304.25**