1. ***Broadly, how would you structure Major League projections to provide distributions of outcomes rather than just point estimates? You do not need to provide code, just high-level structure. [250 words]***
    - To provide distribution outcomes I'd focus on Confidence Intervals (CI), Monte Carlo simulations, and machine learning techniques.
    - Confidence Intervals can provide a range of projections with an associated level of confidence enabling coaches to understand not just the most likely outcomes but the variability around them, providing a more comprehensive view of future possibilities.
    - Monte Carlo in contrast are simulations for dealing with various uncertain variables which could be relevant like player performance and weather conditions. They work by running a large number of simulations with varying inputs and predicting an output. By selecting values randomly from input variable distributions and running numerous simulations, we can obtain a range of likely outcomes based on diverse inputs.
    - In the realm of AI, I'd focus on neural nets (for complex insights) and random forests. Neural networks typically output a single prediction, so to generate a distribution, we could introduce variability, either using dropout layers during testing or by training the network to output probability distribution parameters like mean and standard deviation. This approach would yield not just a likely outcome but also its associated probability distribution. Random forests, comprising multiple decision trees each trained on a different data subset via bootstrap sampling, inherently produce a variety of outputs. Each tree's differing prediction contributes to an overall distribution of outcomes, capturing the uncertainty and variability inherent in sports performance.

2. ***Can you provide a code snippet in your preferred programming language to fit a Bayesian linear regression model to predict a batter's on-base percentage (OBP) based on their batting average (AVG) and walks per plate appearance (BB/PA)? How would you interpret the results of this model and how could it be useful in evaluating a batter's performance?***
    - Statsmodels Approach

```
Import pandas as pd
Import statsmodels.api as sm
From statsmodels.formula.api import glm #Generalized linear model
Import statsmodels.genmod.families as smf

Df = pd.read_csv('data.csv')
Formula = 'OBP ~ AVG + BB_PA' #Assuming I have columns in a dataframe
#with these headers

Model = glm(formula, data = df, family =
smf.Gaussian(sm.families.link.logit)).fit()

print(model.summary())
```

    - Sklearn Approach (bayesian regression isnt exactly available but BayessrianRidge is similar)

```
Import pandas as pd
From sklearn.model_selection import train_test_split
```

```
From skleanr.linear_model import BayesianRidge
From sklearn.metrics import mean_squared_error

Df = pd.read_csv('mydata.csv')
X = df[['AVG, 'BB_PA']]
Y = df['OBP']

X_train, x_test, y_train, y_test = train_test_split(X,y, train_size = 0.7)

Model = BayesianRidge()
model.fit(X_train, y_train)

Y_pred = model.predict(X_test)
Mse = mean_squared_error(y_test, y_pred)

print('mean squared error: ', mse)
print('coefficients: ', model.coef_)
```

- Interpretation: As shown in the sklearn approach I would use mean squared error to assess the accuracy and effectiveness of the model. Other metrics include RMSE, MAE, R-squared, and residual analysis. Selection of these evaluation criterion will depend on data distribution, outliers, etc.
- Usefulness: The model (specifically the coefficients) will help us determine how much each of the two variables (AVG and BB/PA) contribute to OBP. Higher coefficients mean that the variable is associated with an increased OBP, conversely a negative coefficient would imply that higher AVG, BB/PA are associated with lower OBP which would not make sense in this context but is explained for demonstration purposes.

3. ***Rank the pitcher's stuff from 1-3 (1=best, 3=worst)***
   - 1 = Fastball - Good velocity, spin rate, and great vertical movement
     2 = Slider - Good spin rate but could improve that and the horizontal movement by throwing it harder
     3 = Changeup - Throw it slower to get more vertical drop and separation off the fastball velocity
   - ***What adjustments would you recommend to individual pitches or arsenal as a whole***
     i. Find a way to decrease the horizontal movement on the fastball, which ideally helps to increase the spin rate/velocity
     ii. Throw the slider harder
     iii. Find a way to get more depth out of the changeup so it has more separation off the fastball. Potentially add a curveball to play off the high vertical fastball.
4. ***Rank the hitters from most interested to least interested. Explain your reasoning***
   - Most interested = Player A - Great bat to ball skills for in-zone and out-of-zone pitches. Although exit velocity is the lowest (barely) if this player is small (size was not mentioned) then weight/strength gain could naturally improve the exit velocity. This

player clearly has the best approach of the three and it's easier to improve muscular strength and power than to establish a strong consistent approach at the plate. Very few holes in swing.
    - Middle = Player C - Placed second for the obvious power potential. Could improve approach. He chases and misses a lot, but he also swings the most. The low Z-contact% suggests there are some large holes in the swing. Focus on closing holes in swing and decreasing swing rate means high potential.
    - Least interested = Player B - Nothing stands out about this player compared to A or C.
5. **Attached are pitch-level data. Please construct a model predicting the probability of a called strike and explain how you evaluated your model. Please include all code.**
    - See attached solution.