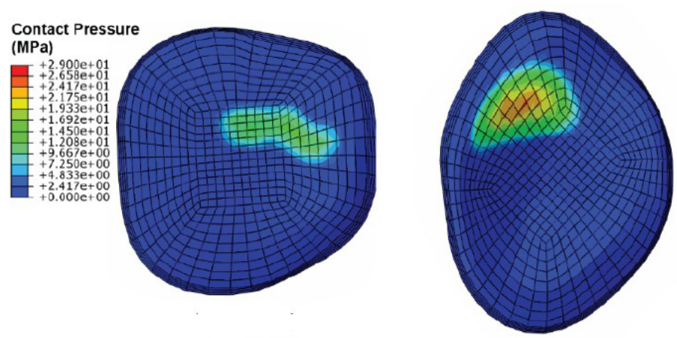
Knee Stress Prediction

Project Sponsor:
Dr. Clare Fitzpatrick, Associate Professor,
Mechanical and Biomedical Engineering



Team AI CBL

Harry Hsu | *Nina Nikitina* | *Stephanie Ball*

CS 481
Fall 2022

1 Introduction

1.1 Problem and Project Statement

This project is purely a research project whose aim is to create and test a machine learning algorithm that can predict knee peak stress/strain with specified accuracy based on the geometry of the knee and loading conditions.

1.2 Operational Environment

Initially, the end product will be used by a research scientist in the Computational Biosciences Laboratory of Mechanical and Biomedical Engineering Department at Boise State University. Probably in the future, the model can also be used by doctors however in this case integration with GUI is required.
Our program will require anaconda to install specific library only work on x86, therefore machine runs the model has to operate with x86 CPU and GPU.

1.3 Functional Requirements

Literature Research
Research project details and similar research to find the best model to implement.

Exploratory Analysis

- We will start with uploading the given data and being able to display an example from the dataset to see what we have to work with.
- Be able to handle poor input data without crashing
- Be able to answer how many observations we have and how many features.
- Plot distributions with a grid of histograms to have a visual understanding of the distributions.

Data Cleaning

- Remove any unnecessary data such as duplicate observations.
- Handle any missing data.
- Combine results dataset with Geometries dataset.

Feature Engineering

- Using our plot distributions and information from data cleaning we can create features that have the most impact.
- Remove any unused features.

Model/models training

- Create a linear regression, Random Forest and Gradient Boosting regressor algorithms.
- Split dataset into training and testing sets with an 80/20 split ratio.
- Train all three models and compare results.
- Choose which model to proceed with.

Verification Result analysis
Using the test dataset and pythons sklearn scoring model we will be able to determine how accurate our model is at predicting future pressure points.

Display Results
Create a mesh using PyVista of the predicted future pressure points

1.4 Assumptions and Limitations

Anaconda env with:
BASICS

- numpy
- scipy
- pandas
- xlrd
- tqdm
- click
- ipython

VISUALIZATION

- matplotlib
- ipympl # interactive matplotlib plots
- seaborn
- pyvista
- meshio=5.0.2
- altair
- yellowbrick

ML, STATS & DEEP LEARNING

- statsmodels
- scikit-learn
- sktime
- tslearn
- xgboost
- catboost
- lightgbm
- pytorch
- tensorflow
- keras
- spacy

1.5 Deliverables

List and describe exactly what will be delivered to the project sponsor, including binaries, source code, documentation, and test results.

- Jupyter notebook that allows running prediction on a user data set.

- Jupyter notebook that demonstrates stress prediction results based on three different models that notebook will be well documented with our findings and specify why decisions were made (i.e. Creating/removing features)

2 Specifications and Analysis

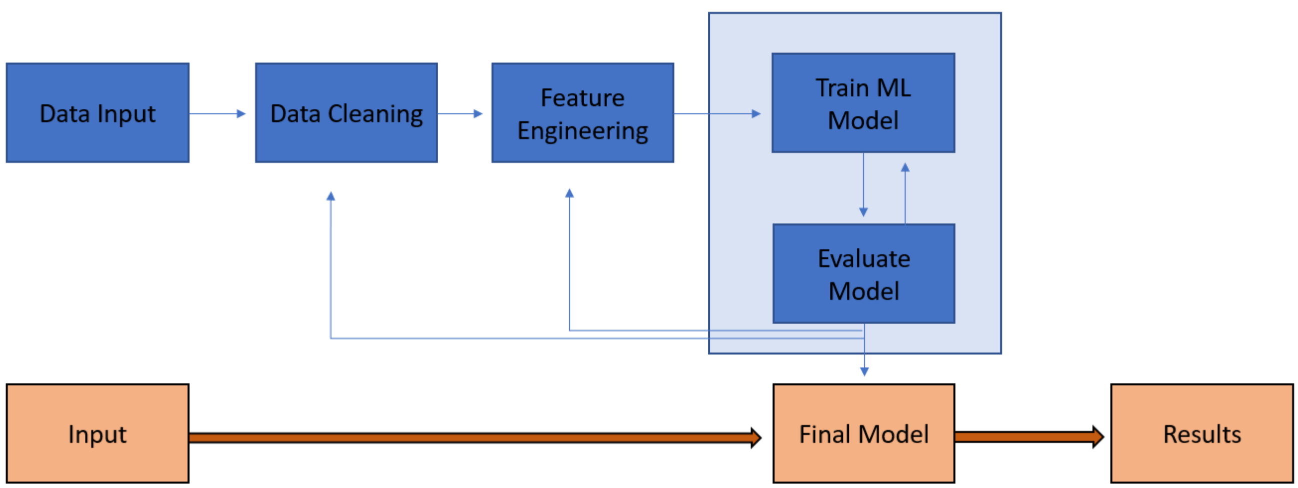
2.1 Proposed Approach

Describe at least one alternate approach that the team considered and present an argument for why it is inferior to the proposed approach.

Our approach is to develop three different Machine Learning Models. We will write this in Python programming language and use the Python libraries described in section 1.4. Once we have trained, tested and compared the different models we will meet with our sponsor to go over our findings and determine which is the best model. We will visually display the results using PyVista that will give us a mesh representation of the predicted stress points in the knee.

We are going to concentrate on classical sed ML models. Additionally to classical approaches, we considered using a mech segmentation technique described in "Schneider, Lisa, et al. "MedmeshCNN-Enabling meshing for medical surface models." *Computer Methods and Programs in Biomedicine* 210 (2021): 106372." however, we decided to leave the model for further researchers and concentrate on widely used ML models in order to deliver solid results and evaluate the capability of ML to solve this kind of scientific problem.

2.2 Architectural Diagram



The diagram above shows our training diagram in blue and the final implementation in orange. It starts with the data input that will then go through the data cleaning process described in sections 1.3. We will then do some feature engineering to create data that will help train the model for best results. Once this is done, we will split the data into a training and testing set and use the training data to train the model. The model will be tested with the test data set and evaluated to determine whether or not we need to go back to clean the data and/or do some more feature engineering. This cycle continues until we find a satisfactory model. With the final model, we will be able to input data and it will return results consisting the high stress points of the knee.

3 Statement of Work

3.1 Related Work

The literature review is supposed to be an important part of the project. One of the similar projects that were realized by a master's student of the LAB is the “Development of a statistical shape-function model of the implanted knee for real-time prediction of joint mechanics”.

3.2 Description of End Products

The end product will contain multiple python files, which process the data and data visualization in modeling and GUI interface. It's designed based on the OOP principles, in which the user can directly run the target part with the specific variable assigned to show the GUI. The end product will contain Jupyter Notebooks with code dedicated to feature engineering and data cleaning and the selection of algorithms to produce machine learning prediction output. Besides the Jupyter notebook and python, we will contain an individual model that was pre-created and saved for predicting outcomes or retraining based on future needs. Machine Learning Model will allow customers to upload a dataset containing information on stress points on a patient's knee, run it through the training model, and get a visual mesh showing the predicted future stress points on a patient's knee.
README file will also be included as part of the end products for the sponsor to understand fully how to start from a brand new laptop to install environment, run the OOP target data visualization, Jupyter notebook tutorial to run the process from data->data clean->modeling->use model for prediction. All of which are included in README.

3.3 Risks and Contingencies

Being a research project, there are many risks involved. When training machine learning models a big risk is whether or not you have clean and accurate data. This weighs heavily on the data cleaning process and feature engineering. If we are not able to combine and find correlation in the different files, we will not be able to create a model that produces predictions better much better than a guess. Because data cleaning and feature engineering will take up most of our time on this project, it is possible we will not know we don’t have accurate results until it is too late. Given our data is produced by linear motion, and the geometry is defined and contained within a certain motion, it is possible to have validation data that may not be able to work much good in the linear back and forth motion, such that the validation data is part of the repeated motions.

4 Project Milestones and Evaluation Criteria

- Milestone 1: Data preparation - week 5
    - Data Visualization(09/20)
    - Data Cleaning
    - Features Engineering
    - Split data for training/testing

- Milestone 2: Model/models training - week 6, 8*

    - Choose training model
    - Create training algorithm
    - Input training data

- Milestone 3: Verification Result analysis - week 7, 9*
    - Use test data to verify results

- Milestone 4: Final Output - week 10
    - Choose best model as the final
    - Present best model to sponsor
    - GUI

* Milestones 2 and 3 are a revolving process and will take going back and forth.

5 Testing and Validation

For testing we will create unit tests that will test how we handle data files with bad input. This will include files with incorrect number of columns or rows, files we don't accept, null values, or values that are not floats.

For validation we will set apart some of the given data for a validation dataset. We can use this data to validate our final models predictions. We will also perform Model Evaluation Matrics throughout the process such as precision, recall, accuracy and F1 score to determine whether or not our model is improving.

A linter will be implemented into our Jupyter Notebook using PyLint to analyze our code and check for errors.