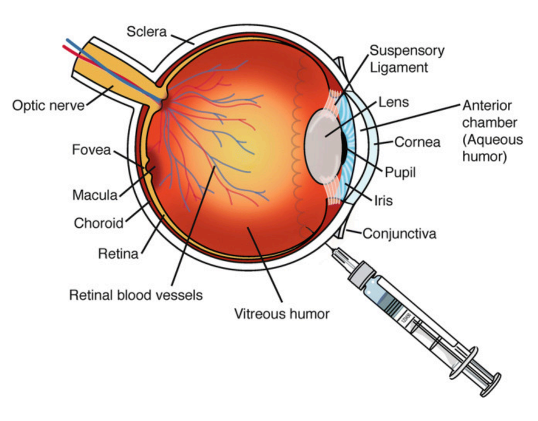
Draft analysis

Name: Samuel Hempelt

Introduction

Diabetic retinopathy is a serious illness, which is expected to affect > 200 million people by the year 2025 [1]. It is an eye desease resulting in blindness for over 10000 people with diabetes per year [2]. In order to help these patients Boehringer Ingelheim is investing in Research and Development of biopharmaceuticals and is screening for new active ingredients, which has the potential to slow or even stop the progression of this desease [3]. A unique characteristic of these medications is the intravitreal application, which means that the drug product is injected directly into the vitreous humor, the gel-like substance inside the eye (see picture below).



Picture: Illustration of the ocular anatomy and intravitreal injection for the treatment of ocular diseases [4]

One major challenge of developing drug products, which are applied intravitreal, is the requirement for the low viscosity of the drug product solution. Viscosity is the measure of how easily a fluid flows; thicker liquids like honey have high viscosity, while thinner ones like water have low viscosity. It reflects the internal resistance of a liquid's molecules to movement or flow. A high viscosity of the drug product solution in the syringe results in a higher injection force necessary to apply the medication into the eye. The European Pharmacopoeia (EP) provides specific guidelines regarding the viscosity of intravitreal applied biopharmaceuticals to ensure safe and effective injection [5].

For this reason the viscosity is a very important measure and is determined several times during the early development stage for every new product. Viscosity is tested under different experiment conditions like temperature and product concentration. In order to reduce development time to the commercial launch of a new drug product and reduce costs for laboratory equipment and personnel, the long term motivation is to predict the viscosity of every new agent without any experiments in the laboratory.

The data set, which will be explored in this work consists of viscosity data, whereas each observation of the data set corresponds to one measurement value. The data was collected as part of a characterization study for various biopharmaceutical products. These products consist of different types of proteins (IgG2, IgG4, Knob/Hole, DoppelMab), which have different characteristics like molecular weight, isoelectic point or extinction coefficient. In order to determine the effect of product concentration on the viscosity, each product was measured at two different concentrations (10 mg/mL, 62.5 mg/mL). Furthermore, viscosity was measured at different temperatures (2°C - 40°C) to assess the impact of temperature variations. The data set consists of the following variables:

Name	Description	Role	Type	Format
viscosity_mPas	Measured viscosity, of the sample in mPas	response	numeric	float
replicate	Number of replicate. Within each measurement, two individual measurements were conducted as technical replicates	ID	numeric	int
entered_on	The date on which the measurement was conducted	predictor	numeric	date
instrument	Instrument, which was used to measure the viscosity	predictor	nominal	category
temperature_c	The temperature at which the measurement was conducted	predictor	numeric	float
product_concentration_mg_mL	Concentration of the product in the aqueous solution in mg/mL	predictor	numeric	float
product	Internal product name as a unique code	ID	nominal	category
protein_format	Protein format of the investigated product	predictor	nominal	category
molecular_weight_kda	Molecular weight of the investigated product in kDa. A measure of the size of the protein	predictor	numeric	float
extinction_coefficient_l_molcm	Extinction coefficient of the investigated product in $L \cdot mol^{-1} \cdot cm^{-1}$. A measure of the light absorption ability of the molecule	predictor	numeric	float
isoelectric_point	Isoelectric point of the investigated product. A measure of the charge of the molecule	predictor	numeric	float

In this work the impact of different experiment conditions on the measured viscosity value of the drug product was examined. Different variables like the temperature, the product concentration, and the molecular weight are considered as possible predictors and likely have an impact on the response variable. After analysing the relationship between these explanatory variables and the viscosity a model will be fitted, which makes further investigations possible.

Setup

In [566…
```
import pickle
import os
from datetime import datetime
import subprocess
from pathlib import Path

import pandas as pd
import altair as alt
import numpy as np

from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import cross_val_score

from sklearn.metrics import mean_squared_error
```

Data

The underlying measurements were conducted by laboratory personnel within the formulation development department at Boehringer Ingelheim. The measurement values were documented in the internal Laboratory Information Management System (LIMS) along with additional detailed experimental and contextual information, which also contain all explanatory data like the product characteristics etc. The data is stored in an Oracle SQL database and was extracted using targeted SQL queries.

Import data

In [567…
```
# Import data from the csv-file "viscosity_data.csv"
df = pd.read_csv("viscosity_data.csv", sep=";")
```

Data structure

In [568…
```
df.head()
```

Out[568…

	viscosity_mPas	replicate	entered_on	instrument	temperature	product_concentration_mg_mL	product	protein_format	molecular_weight_kDa	extincti
0	3.93	1	15.03.2019	VISCOSIMETER_02	2	10.0	BI655300	IgG2	148830	
1	4.28	2	16.03.2019	VISCOSIMETER_02	2	10.0	BI655300	IgG2	148830	
2	3.42	1	15.03.2019	VISCOSIMETER_02	5	10.0	BI655300	IgG2	148830	
3	3.69	2	15.03.2019	VISCOSIMETER_02	5	10.0	BI655300	IgG2	148830	
4	2.89	1	15.03.2019	VISCOSIMETER_02	10	10.0	BI655300	IgG2	148830	

In [569…
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 502 entries, 0 to 501
Data columns (total 11 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   viscosity_mPas                  502 non-null    float64
 1   replicate                       502 non-null    int64
 2   entered_on                      502 non-null    object
 3   instrument                      502 non-null    object
 4   temperature                     502 non-null    int64
 5   product_concentration_mg_mL     502 non-null    float64
 6   product                         502 non-null    object
 7   protein_format                  502 non-null    object
 8   molecular_weight_kDa            502 non-null    int64
 9   extinction_coefficient_L_molcm  502 non-null    float64
 10  isoelectric_point               502 non-null    float64
dtypes: float64(4), int64(3), object(4)
memory usage: 43.3+ KB
```

Data corrections

According to the literature there are three explanatory variables, which have an impact on the viscosity of the solution. According to the Arrhenius equation, a higher temperature generally decreases viscosity, because molecular movement increases, reducing intermolecular interactions [6]. Also the product concentration might have an impact on the viscosity, because molecules in solution interact more frequently, leading to increased resistance to flow [7] Additionally larger and more complex proteins, such as aggregates or conjugated proteins, tend to increase solution viscosity due to their size and interaction with other molecules in the solution [8], which indicates that the moleculare weight of the product might have an impact on the measured viscosity. According to literature the relationship of all three explanatory variables and the target variable is exponentell. In order to use the linear regression the Log Transformation is used. Therefore the target variable 'viscosity_mpas' is normalized by calculating the base-10 logarithm of the variable.

In [570…
```
# Make sure column names are lower case and eliminate spaces
df.columns = df.columns.str.lower()
```

In [ ]:
```
# Log-transformation of the target variable
df['log_viscosity_mpas'] = np.log(df['viscosity_mpas'])
```

In [572…
```
# For a better overview data set is reduced to the most interesting variables we want to examine
df = df.iloc[0:502,[4,5,8,11]]
```

In [573…
```
df.head()
```

Out[573…

	temperature	product_concentration_mg_ml	molecular_weight_kda	log_viscosity_mpas
0	2	10.0	148830	1.368639
1	2	10.0	148830	1.453953
2	5	10.0	148830	1.229641
3	5	10.0	148830	1.305626
4	10	10.0	148830	1.061257

Variable lists

In [574…
```
# define outcome variable as y_label
y_log_label = 'log_viscosity_mpas'

# select features
X = df[['temperature','product_concentration_mg_ml','molecular_weight_kda']]

# create response
y_log = df[y_log_label]

# Create list with numeric features
list_numeric =['temperature','product_concentration_mg_ml','molecular_weight_kda']
```

Data splitting

In [575…
```
# use a test size of 0,2 and random state 42
X_train, X_test, y_train, y_test = train_test_split(X, y_log, test_size=0.2, random_state=42)

# use your training data to make a pandas dataframe
df_train = pd.DataFrame(X_train.copy())

# add your training labels to the data
df_train = df_train.join(pd.DataFrame(y_train))

df_train.head(5)
```

Out[575…

	temperature	product_concentration_mg_ml	molecular_weight_kda	log_viscosity_mpas
423	25	62.5	148977	1.401183
19	2	62.5	148830	1.905088
323	40	62.5	149601	1.131402
333	20	10.0	149683	1.160021
56	5	10.0	149610	0.959350

Analysis

Descriptive statistics

In [576…
```
df.describe().T
```

Out[576…

	count	mean	std	min	25%	50%	75%	max
temperature	502.0	20.203187	12.610650	2.000000	10.000000	20.000000	30.000000	40.000000
product_concentration_mg_ml	502.0	36.354582	26.275976	10.000000	10.000000	62.500000	62.500000	62.500000
molecular_weight_kda	502.0	161211.103586	22382.413980	146286.000000	148783.000000	149601.000000	155089.000000	206428.000000
log_viscosity_mpas	502.0	1.173634	0.580932	-0.616186	0.788457	1.152152	1.514577	3.327551

Exploratory data analysis

In order to visualize the relationship between the explanatory variables and the response variable, charts were created with the training data. According to literature a linear correlation of all three parameters (temperature, product concentration, molecular weight) and the logarithm of the viscosity is expected. Although all three predictor variables take continuous numerical values, the dataset contains only a specific number of defined values for the independent variables. There exist only nine values of the temperature (2°C, 5°C, 10°C, 15°C, 20°C, 25°C, 30°C, 35°C, 40°C), two values of the product concentration (10 mg/mL, 62.5 mg/mL) and 14 values of the molecular weight. Therefore box plots were chosen to visualize the distribution of the response variable within these values of the predictor variables.

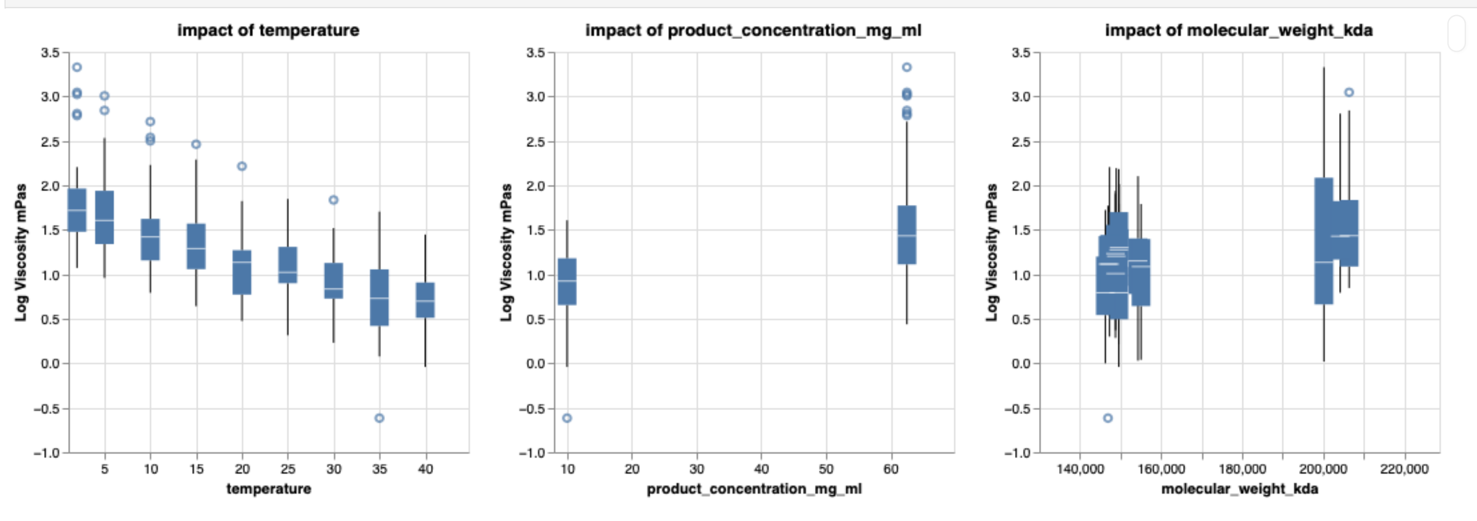
In [577…
```
charts = []  # List, to store all charts

for x in list_numeric:
    boxplot = (
        alt.Chart(df_train)
        .mark_boxplot()
        .encode(
            x=alt.X(x, title=x, scale=alt.Scale(domain=[0.9*df[x].min(), 1.1* df[x].max()])),
            y=alt.Y('log_viscosity_mpas', title=' Log Viscosity mPas')
        )
        .properties(
            title=f'impact of {x}',
            width=300,
            height=300
        )
    )
    charts.append(boxplot)

    final_chart = alt.hconcat(*charts)

final_chart
```

Out[577…



In the left diagram, the relationship between the logarithm of the measured viscosity in mPas and the temperature in °C is shown. A clear negative trend is evident from the individual data points. As temperature increases, the logarithm of the measured viscosity decreases, indicating a negative association between the two variables. Although most of the outliers in the graph are above the boxes, the distribution of the response variable appears to be symmetric. This suggests that the relationship between the two variables is linear when the viscosity is logarithmically transformed, and the relationship between temperature and viscosity (without normalization) would follow an exponential pattern. These findings are consistent with the literature, which states that the viscosity of a fluid decreases exponentially as temperature rises [6].

The centered diagram illustrates the relationship between the logarithm of the measured viscosity (in mPas) and the product concentration (in mg/mL). Although only two values of the predictor variable (10 mg/mL and 62.5 mg/mL) were analyzed, a clear positive correlation between the logarithm of the viscosity and the product concentration is evident. The median positions within the boxes indicate that the distribution of the response variable appears to be symmetric. This symmetry suggests that the relationship between the product concentration and the logarithmically transformed viscosity is linear. Conversely, the relationship with the untransformed viscosity would follow an exponential pattern. These findings align with the literature, which reports that the viscosity of protein solutions increases exponentially with rising protein concentration [7].

The relationship between the logarithm of the measured viscosity (in mPas) and the molecular weight (in kDa) of the product, as shown in the right graph, is less clear. Despite the dataset containing 14 different molecular weights, the predictor variable is not evenly distributed along the x-axis. The graph reveals that the molecular weights can be grouped into two main clusters: a smaller cluster between 140,000 kDa and 160,000 kDa, and a larger cluster between 200,000 kDa and 210,000 kDa. The logarithmically transformed viscosity of the heavier molecules appears to be higher than that of the smaller molecules, suggesting a positive association between molecular weight and logarithmically transformed viscosity.

Relationships

In order to quantify the relationship between the response variable and the predictor variables in the data set pairwise corelation coefficients between all variables were computed

In [578…
```
# for numeric variables pearson correlation coefficient is appropriate
corr = df.corr(method='pearson').round(2)
corr_blues = corr.style.background_gradient(cmap='Blues')
corr_blues
```

Out[578…

	temperature	product_concentration_mg_ml	molecular_weight_kda	log_viscosity_mpas
temperature	1.000000	0.000000	0.000000	-0.630000
product_concentration_mg_ml	0.000000	1.000000	-0.000000	0.520000
molecular_weight_kda	0.000000	-0.000000	1.000000	0.310000
log_viscosity_mpas	-0.630000	0.520000	0.310000	1.000000

In [579…
```
# inspect correlation between response and predictors
corr_list = corr[y_log_label].sort_values(ascending=False)
corr_list
```

Out[579…
```
log_viscosity_mpas             1.00
product_concentration_mg_ml    0.52
molecular_weight_kda           0.31
temperature                   -0.63
Name: log_viscosity_mpas, dtype: float64
```

The calculated correlation coefficients confirm the statements in the above charts of the exploratory data analysis. The highest impact on the response variable has the temperature with a negative correlation coefficient of -0.63 followed by the product concentration with an positive correlation coefficient of 0.52. Although the molecular weight has only a moderate positiv effect on the target variable, this parameter will also be inclued into the calculation of the linear regression model.

Model

Select model

In [580…
```
reg = LinearRegression()
```

Training and validation

In [581…
```
# cross-validation with 5 folds
scores = cross_val_score(reg, X, y_log, cv=5, scoring='neg_mean_squared_error') * -1

# store cross-validation scores (we call the column "lr" for "linear regression")
df_scores = pd.DataFrame({'lr': scores})

# reset index to match the number of folds
df_scores.index += 1

# print nice looking dataframe
df_scores.style.background_gradient(cmap='Blues')
```

Out[581…

	lr
1	0.067927
2	0.088753
3	0.156927
4	0.094414
5	0.103454

In [582…
```
# calculate statistics
df_scores.describe().T
```

Out[582…

	count	mean	std	min	25%	50%	75%	max
lr	5.0	0.102295	0.033214	0.067927	0.088753	0.094414	0.103454	0.156927

The values, which were obtained from the 5-fold cross-validation represent the mean squared error for each of the 5 folds. The values from the cross-validation show that the regression model performs well overall, but there are some variations between the individual folds. It would be useful to further investigate whether there are specific features that cause the model to perform worse in certain folds, and how you can improve the model's performance overall.

Fit model

In [583…
```
reg.fit(X_train, y_train)
```

Out[583…
```
▾ LinearRegression
LinearRegression()
```

In [584…
```
# Intercept
reg.intercept_
```

Out[584…
```
np.float64(0.07207553781356668)
```

In [585…
```
# Coefficient
reg.coef_
```

Out[585…
```
array([-2.97578446e-02,  1.14557436e-02,  7.98704230e-06])
```

Evaluation on test set

In [586…
```
#Prediction and evaluation
y_pred = reg.predict(X_test)
```

In [587…
```
#Retransformation of the prediction
y_test_exp = np.exp(y_test)
y_pred_exp = np.exp(y_pred)
```

In [588…
```
mse = mean_squared_error(y_test_exp, y_pred_exp)
print("Mean Squared Error after reverse log transformation:", mse)
```

```
Mean Squared Error after reverse log transformation: 1.8713359627143917
```

Save model

Save your model in the folder `models/`. Use a meaningful name and a timestamp.

In [589…
```
# timestamp and name
now = datetime.now()
timestamp = now.strftime("%Y%m%d%H%M%S")
model_name = timestamp + "_viscosity_model.pkl"

# give out target directory
repo_directory = subprocess.check_output(["git", "rev-parse", "--show-toplevel"]).strip().decode()
directory = repo_directory + "/models"

# complete path for file
model_path = os.path.join(directory, model_name)

# save model
with open(model_path, "wb") as file:
    pickle.dump(reg, file)

print(f"Model was saved in {model_path} ")
```

```
Model was saved in /Users/snowwhite/Desktop/DataAnalyticswithStatistics/Project/project/models/20241216163517_viscosity_model.pkl
```

Conclusions

The model demonstrates a relatively low error (MSE = 1.87), suggesting reasonable prediction accuracy. This corresponds to an average deviation of about $\sqrt{1.87} \approx 1.37$ (in the same units as the target variable) between predicted and actual values. However, the adequacy of this error depends largely on the target variable's scale.

With the target variable ranging from approximately 0.54 to 27.87, an average error of 1.37 is relatively small. Notably, in some cases, the difference between technical replicates (measurements under identical conditions) exceeds the model's average error, highlighting its potential utility.

To enhance the model further, additional variables in the dataset, such as the extinction coefficient, isoelectric point, or protein format of the molecule, should be investigated and incorporated into the regression model where relevant.

Bibliography

[1] Karin S Coyne, Mary Kay Margolis, Tessa Kennedy-Martin, Timothy M Baker, Ronald Klein, Matthew D Paul, Dennis A Revicki, The impact of diabetic retinopathy: perspectives from patient focus groups, Family Practice, Volume 21, Issue 4, August 2004, Pages 447–453.

[2] Donald S. Fong, Frederick L. Rerres, Lloyd P. Aiello, Ronald Klein, Diabetic Retinopathy, Diabetes Care, Volume 27, Number 10, October 2004

[3] Produkt Portfolio Boehringer Ingelheim, 2023, https://unternehmensbericht.boehringer-ingelheim.de/2023/download/BOE_GB23_Produktportfolio_DE_safe.pdf

[4] Parenky AC, Wadhwa S, Chen HH, Bhalla AS, Graham KS, Shameem M. Container Closure and Delivery Considerations for Intravitreal Drug Administration. AAPS PharmSciTech. 2021 Mar 11;22(3):100. doi: 10.1208/s12249-021-01949-4. PMID: 33709236; PMCID: PMC7952281.

[5] European Directorate for the Quality of Medicines & HealthCare. European Pharmacopoeia. 10th ed., Council of Europe, 2020. www.edqm.eu/en/european-pharmacopoeia-pharmacopoeia-europe.

[6] Arrhenius S. The Viscosity of Solutions. Biochem J. 1917 Aug;11(2):112-33. doi: 10.1042/bj0110112. PMID: 16742728; PMCID: PMC1258811.

[7] Wozniak, Spencer, and Michael Feig. "Diffusion and Viscosity in Mixed Protein Solutions." The Journal of Physical Chemistry B, vol. 128, no. 47, 2024

[8] Woldeyes, M. A., Josephson, L. L., Leiske, D. L., Galush, W. J., Roberts, C. J., & Furst, E. M. (2018). Viscosities and protein interactions of bispecific antibodies and their monospecific mixtures. Molecular Pharmaceutics, 15(10), 4252-4261.