Module 1

Team Members:

Max Calcoen

Medha Tadavarthi

Project Title:

How does your level of education affect Alzheimer's disease pathology and Tau levels?

Project Goal:

This project seeks to understand the correlation between Alzheimer's disease and education level.

Disease Background:

Fill in information about 11 bullets:

- Prevalence & incidence (source)
  - Approximately 11% of the U.S. population age 65 and older is affected by Alzheimer's disease
  - The annual incidence of Alzheimer's disease in the U.S. is estimated at 0.4% among individuals age 65 to 74 years, 3.2% among those aged 75 to 84 years, and 7.6% among those aged 85 years and older
- Economic burden (source)
  - The projected economic burden of Alzheimer's disease in the U.S. is expected to reach $781 billion by 2025
- Risk factors (genetic, lifestyle) (source)
  - Older age and genetic predisposition are the primary risk factors for Alzheimer's disease
- Societal determinants (source, source 2)
  - Education, income, access to health care, discrimination, and social isolation contribute to disparities in Alzheimer's risk
  - Black adults aged 65 and older are nearly twice as likely as White adults of the same age group to develop Alzheimer's or other dementias
- Symptoms (source)
  - Cognitive symptoms include memory loss, difficulty concentrating, and impaired thinking.

    - Emotional and behavioral symptoms include depression, loss of interest in activities, social withdrawal, mood swings, distrust, anger or aggression, changes in sleeping habits, wandering, loss of inhibitions, and delusions.
- Diagnosis (source)
    - no single test to evaluate alzheimer's or other dementia
    - medical history: history of cognitive and behavioral changes, family history of alzheimer's or other dementia
    - physical exam, diagnostic tests: diet, medications, physicals, lab test
    - neurological exam: reflexes, coordination, eye movement, speech, sensation
    - cognitive, functional and behavioral tests: AD8, SLUMS, IQCODE, FAQ, MMSE
    - computerized tests and devices: growing area. includes ANAM, CANTAB, CognICA, Cognigram, Cognivue, Cognision (digital cognitive testing tools)
    - depression screen, mood assessment: sense of wellbeing
    - genetic testing: certain genes increase risk of alzheimer's and other rare genes (1% or less) that directly cause alzheimer's
    - brain imaging: MRI or CT scanes rule out other conditions, or sometimes look directly at beta-amyloid
    - cerebrospinal fluid: biomarkers (tau, beta-amyloid, neurofilament light) appear in CSF before brain, still evolving test
    - blood test: evolving method
- Standard of care treatments (& reimbursement) (source)
    - no cure, but various drug and non-drug options for treatment
    - drugs that treat symptoms: cognitive symptoms (chrlinesterase inhibitors, glutamate regulators), non-cognitive symptoms (orexin inhibitors), atypical antipsychotics (target serotonin and dopamine chemical pathways)
    - drugs that change disease progression: 2 FDA-approved treatments (both anti-amyloid antibody IV infusions)
    - as with many neurodegenerative disease, early diagnosis is important (some treatments only effective in early stages of disease). AD is a progressive brain disease: worsens over time (asymptomatic -> mild cognitive impairment -> mild -> moderate -> severe)
- Disease progression & prognosis (source)
    - 3 stages: early (mild), middle (moderate), and late (severe)
    - on average, a person lives 4-8 years after diagnosis
    - early stage: may function independently. feel as if they have memory lapses (forgetting words, familiar objects, names, etc)
    - middle stage: longest stage, lasts many years. dementia symptoms progress, including memory loss, confusion, trouble controlling bladder/bowels, personality / behavioral changes
    - late stage: lose ability to respond to environment, control movement, communication becomes difficult. require assistance with daily personal care, lose awareness, difficulty communicating
- Continuum of care providers (source)
    - depends on location (country > state > region > city)

    - primary and specialized medical providers provide timely and accurate diagnosis, neurologists, psychiatrists, neuropsychologists offer early detection and specialized treatment services
    - care coordination and transition services: care coordinators oversee transitions between stages of care (diagnosis to long term management)
    - long term support and residential care: assisted-living facilities, continuing care retirement communities, nursing homes (hopefully with dementia-specific protocols)
    - support for caregivers and families (educational materials, culturally sensitive support, community based programs)
    - public health and policy level
- Biological mechanisms (anatomy, organ physiology, cell & molecular physiology) (source)
    - amyloid-beta (Aβ) protein is the principal component of AD-associated amyloid plaques, produced by type I transmembrane amyloid precursor protein (APP)
    - Aβ believed to play a role in neuroprotection, trophic support, and cell adhesion
    - dysregulated APP processing may contribute to AD pathogenesis by elevating Aβ production, leading to build up and aggregation into neurotoxic forms
    - multiple genes associated with APP processing are risk factors for FAD (familial AD)
    - Aβ aggregates are assmembed from Aβ monomers (oligomeric Aβ), further aggregates and elongates into insoluble fibrillar assemblies comprising β-strand repeats
    - Aβ oligomers rather than Aβ fibrils are neurotoxic
    - oAβ disrupts intracellular calcium balance, impair mitochondria dysfunction, and induce the production of reactive oxygen species (ROS),leading to neuronal apoptosis and cell death
- Clinical Trials/next-gen therapies (source)
    - (same as standard of care)
    - drugs that treat symptoms: cognitive symptoms (chrlinesterase inhibitors, glutamate regulators), non-cognitive symptoms (orexin inhibitors), atypical antipsychotics (target serotonin and dopamine chemical pathways)
    - drugs that change disease progression: 2 FDA-approved treatments (both anti-amyloid antibody IV infusions)

Data-Set:

Data pulled from Integrated multimodal cell atlas of Alzheimer's disease

The available dataset contains Luminex multiplex immunoassay measurements (protein concentrations of biomarkers amyloid-beta and tau) from postmortem brain tissue, reported in pg/mL. These measurements are paired with donor metadata (age, sex, diagnosis, cohort, disease severity) that allow stratification across the Alzheimer's disease spectrum. Data were collected from human brain tissue donors in the SEA-AD project and processed under uniform protocols to ensure comparability across individuals

Data Analysis:

```python
# view.ipynb (exported as py)
# %%
import pandas as pd

luminex = pd.read_csv("../data/UpdatedLuminex.csv")
metadata = pd.read_csv("../data/Metadata_v2.csv")

# %%
display(luminex.head())
display(metadata.head())

# %%
# only merge if merge file doesn't exist
try:
    merged = pd.read_csv("../data/merged.csv")
    display(merged.head())
except FileNotFoundError:
    merged = pd.merge(luminex, metadata, on="Donor ID")
    display(merged.head())
    merged.to_csv("../data/merged.csv", index=False)

# %%
# print(luminex.columns)
# print(metadata.columns)
print(merged.columns)

# %%
# TODO: isolation forest to remove outliers


# %%
# remove outlier in tTAU
merged_filtered = merged[merged["tTAU pg/ug"] < 5000]
display(merged["tTAU pg/ug"].describe())
display(merged_filtered["tTAU pg/ug"].describe())

merged_filtered.to_csv("../data/merged_filtered.csv", index=False)

# %%
# double check APOE genotype column
merged_filtered["APOE Genotype"]


# %%
# simple wrapper for donor record
class Patient:
    def __init__(self, record):
        self.record = record
        self.donor_id = record.get("Donor ID")
```

```
        self.apoe_genotype = record.get("APOE Genotype")

    def __repr__(self):
        return f"Patient(donor_id={self.donor_id},
APOE={self.apoe_genotype})"


# %%
# create patient objects
patients = [Patient(row) for _, row in merged_filtered.iterrows()]
print(f"Created {len(patients)} patient objects")
# display head
patients[:5]

# main.ipynb (exported as py)
# %%
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from scipy import stats
from statsmodels.formula.api import ols
import statsmodels.api as sm

luminex = pd.read_csv("../data/UpdatedLuminex.csv")
metadata = pd.read_csv("../data/Metadata_v2.csv")
df = pd.read_csv("../data/merged_filtered.csv")

# %%
# data cleaning and preprocessing / feature engineering
df["Thal (n)"] = df["Thal"].str.split(" ").str[1].astype(int)

df["ABeta ratio"] = df["ABeta42 pg/ug"] / df["ABeta40 pg/ug"]
df["TAU ratio"] = df["pTAU pg/ug"] / df["tTAU pg/ug"]
education_order = [
    "High School",
    "Trade School/ Tech School",
    "Bachelors",
    "Graduate (PhD/Masters)",
    "Professional",
]
df["Highest level of education"] = pd.Categorical(
    df["Highest level of education"], categories=education_order,
ordered=True
)
# condense into 2 groups, high school and post-secondary
df["Post-secondary"] = pd.Categorical(df["Highest level of
education"].apply(
    lambda x: "High School" if x == "High School" else "Post-
Secondary"
), categories=["High School", "Post-Secondary"], ordered=True)
```

```
# condense into 2 groups, e4+ and e4-
df["APOE Genotype 4"] = df["APOE Genotype"].apply(
    lambda x: "e4+" if "4" in x else "e4-"
)

# %%
# define x and y for analysis
x_anova = "Highest level of education"
x_lin_reg = "Years of education"
x_t = "Post-secondary"
y = "TAU ratio"

# %%
# initial visualization
# bar plot multiple groups
plt.figure(figsize=(10, 6))
sns.barplot(data=df, x=x_anova, y=y, errorbar="se")
plt.ylabel(y)
plt.xlabel(x_anova)
plt.title(f"{y} by {x_anova}")
plt.tight_layout()
plt.show()

# bar plot two groups
plt.figure(figsize=(10, 6))
sns.barplot(data=df, x=x_t, y=y, errorbar="se")
plt.ylabel(y)
plt.xlabel(x_t)
plt.title(f"{y} by {x_t}")
plt.tight_layout()

# scatter
plt.figure(figsize=(10, 6))
sns.scatterplot(data=df, x=x_lin_reg, y=y)
plt.ylabel(y)
plt.xlabel(x_lin_reg)
plt.title(f"{y} by {x_lin_reg}")
plt.tight_layout()
plt.show()

# %%
# run ANOVA for every group in x
try:
    model = ols(f'Q("{y}") ~ Q("{x_anova}")', data=df).fit()
except Exception as e:
    print("likely bad columns specified")
    print(f"Error in model fitting: {e}")
anova_table = sm.stats.anova_lm(model, typ=2)
print(anova_table)
```

```
# %%
# run Tukey's HSD test for post-hoc analysis (anova)
from statsmodels.stats.multicomp import pairwise_tukeyhsd
tukey = pairwise_tukeyhsd(endog=df[y], groups=df[x_anova], alpha=0.05)
print(tukey)

# %%
# t-test between the two groups

group1_name = "High School"
group2_name = "Post-Secondary"

group1 = df[df[x_t] == group1_name][y].dropna()
group2 = df[df[x_t] == group2_name][y].dropna()

t_stat, p_value = stats.ttest_ind(group1, group2)
print(
    f"T-test between {group1_name} and {group2_name} for {y}: t-
statistic = {t_stat}, p-value = {p_value}"
)

# %%
# linear regression
slope, intercept, r_value, p_value, std_err = stats.linregress(
    df[x_lin_reg].dropna(), df[y].dropna()
)
print(
    f"Linear regression of {y} on {x_lin_reg}: \nslope = {slope},
intercept = {intercept}, r-squared = {r_value**2}, p-value =
{p_value}"
)

# %%
# graph linear regression
plt.figure(figsize=(10, 6))
sns.regplot(data=df, x=x_lin_reg, y=y, line_kws={"color": "red"})
plt.ylabel(y)
plt.xlabel(x_lin_reg)
plt.title(f"Linear regression of {y} on {x_lin_reg}")
plt.tight_layout()
plt.show()

# %% [markdown]
# the below blocks are not related to objectives of assignment but
contain more interesting investigations

# %%
cols_to_drop = [
    "Donor ID",
    "Primary Study Name",
```

```
    "Secondary Study Name",
    "Age at Death",
    "Cognitive Status",
    "Age of onset cognitive symptoms",
    "Age of Dementia diagnosis",
    "Known head injury",
    "Have they had neuroimaging",
    "Consensus Clinical Dx (choice=Alzheimers disease)",
    "Consensus Clinical Dx (choice=Alzheimers Possible/ Probable)",
    "Consensus Clinical Dx (choice=Ataxia)",
    "Consensus Clinical Dx (choice=Corticobasal Degeneration)",
    "Consensus Clinical Dx (choice=Control)",
    "Consensus Clinical Dx (choice=Dementia with Lewy Bodies/ Lewy
Body Disease)",
    "Consensus Clinical Dx (choice=Frontotemporal lobar
degeneration)",
    "Consensus Clinical Dx (choice=Huntingtons disease)",
    "Consensus Clinical Dx (choice=Motor Neuron disease)",
    "Consensus Clinical Dx (choice=Multiple System Atrophy)",
    "Consensus Clinical Dx (choice=Parkinsons disease)",
    "Consensus Clinical Dx (choice=Parkinsons Cognitive Impairment -
no dementia)",
    "Consensus Clinical Dx (choice=Parkinsons Disease Dementia)",
    "Consensus Clinical Dx (choice=Prion)",
    "Consensus Clinical Dx (choice=Progressive Supranuclear Palsy)",
    "Consensus Clinical Dx (choice=Taupathy)",
    "Consensus Clinical Dx (choice=Vascular Dementia)",
    "Consensus Clinical Dx (choice=Unknown)",
    "Consensus Clinical Dx (choice=Other)",
    "If other Consensus dx, describe",
    "Last CASI Score",
    "Interval from last CASI in months",
    "Last MMSE Score",
    "Interval from last MMSE in months",
    "Last MOCA Score",
    "Interval from last MOCA in months",
    "PMI",
    "Rapid Frozen Tissue Type",
    "Ex Vivo Imaging",
    "Fresh Brain Weight",
    "Brain pH",
    "Overall AD neuropathological Change",
    "Thal",
    "Braak",
    "CERAD score",
    "Overall CAA Score",
    "Highest Lewy Body Disease",
    "Total Microinfarcts (not observed grossly)",
    "Total microinfarcts in screening sections",
```

```
    "Atherosclerosis",
    "Arteriolosclerosis",
    "LATE",
    "RIN",
    "Severely Affected Donor",
    "Thal (n)"
]

# %%
# random forest regression using all features
from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, r2_score
# select features and target
features = df.drop(columns=cols_to_drop)
target = df[y]
# one-hot encode categorical variables
features = pd.get_dummies(features, drop_first=True)
# split into training and testing sets, set seed for reproducibility
X_train, X_test, y_train, y_test = train_test_split(features, target,
test_size=0.2, random_state=42)
# train the model
rf = RandomForestRegressor(n_estimators=100, random_state=10)
rf.fit(X_train, y_train)
# make predictions
y_pred = rf.predict(X_test)
# evaluate the model
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)
print(f"Random Forest Regressor MSE: {mse}, R2: {r2}")
# feature importance
feature_names = features.columns
feature_importances = pd.Series(rf.feature_importances_,
index=feature_names).sort_values(ascending=False)
print("Feature importances:")
print(feature_importances.head(20))

# %%
# now drop obvious correlated features and rerun
cols_to_drop_2 = cols_to_drop + [
    "pTAU pg/ug",
    "tTAU pg/ug",
    "ABeta ratio",
    "ABeta40 pg/ug",
    "ABeta42 pg/ug",
    "TAU ratio",
    "ABeta ratio",
]
# select features and target
```

```
features = df.drop(columns=cols_to_drop_2)
target = df[y]
# one-hot encode categorical variables
features = pd.get_dummies(features, drop_first=True)
# split into training and testing sets, set seed for reproducibility
X_train, X_test, y_train, y_test = train_test_split(
    features, target, test_size=0.2, random_state=42
)
# train the model
rf = RandomForestRegressor(n_estimators=100, random_state=42)
rf.fit(X_train, y_train)
# make predictions
y_pred = rf.predict(X_test)
# evaluate the model
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)
print(f"Random Forest Regressor MSE: {mse}, R2: {r2}")
# feature importance
feature_names = features.columns
feature_importances = pd.Series(
    rf.feature_importances_, index=feature_names
).sort_values(ascending=False)
print("Feature importances:")
print(feature_importances.head(20))

# %%
# ALSO: just to show a good linear regression example

df_copy_lin_reg = df.copy()
x_lin_reg_2 = "Age of Dementia diagnosis"
y_lin_reg_2 = "Age of onset cognitive symptoms"

# remove NaN rows for the two columns of interest
df_copy_lin_reg = df_copy_lin_reg[
    df_copy_lin_reg[x_lin_reg_2].notna() &
df_copy_lin_reg[y_lin_reg_2].notna()
]

slope, intercept, r_value, p_value, std_err = stats.linregress(
    df_copy_lin_reg[x_lin_reg_2].dropna(),
df_copy_lin_reg[y_lin_reg_2].dropna()
)
print(
    f"Linear regression of {y_lin_reg_2} on {x_lin_reg_2}: \nslope =
{slope}, intercept = {intercept}, r-squared = {r_value**2}, p-value =
{p_value}"
)

# %%
# graph
```

```
plt.figure(figsize=(10, 6))
sns.regplot(data=df_copy_lin_reg, x=x_lin_reg_2, y=y_lin_reg_2,
line_kws={"color": "red"})
plt.ylabel(y_lin_reg_2)
plt.xlabel(x_lin_reg_2)
plt.title(f"Linear regression of {y_lin_reg_2} on {x_lin_reg_2}")
plt.tight_layout()
plt.show()

# %%
# elastic net regression (ridge + lasso)
from sklearn.linear_model import ElasticNet
from sklearn.preprocessing import StandardScaler

elastic_alpha = 0.5
# reuse feature list that drops highly correlated biomarkers
elastic_features = df.drop(columns=cols_to_drop_2)
elastic_target = df[y]
elastic_features = pd.get_dummies(elastic_features, drop_first=True)

X_train, X_test, y_train, y_test = train_test_split(
    elastic_features, elastic_target, test_size=0.2, random_state=42
)

scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

elastic_net = ElasticNet(alpha=elastic_alpha, l1_ratio=0.5,
random_state=42, max_iter=10000)
elastic_net.fit(X_train_scaled, y_train)

y_pred = elastic_net.predict(X_test_scaled)
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)
print(f"Elastic Net (alpha={elastic_alpha}) MSE: {mse:.4f}, R2:
{r2:.4f}")

coefficients = pd.Series(elastic_net.coef_,
index=elastic_features.columns)
non_zero_coeffs = coefficients[coefficients !=
0].sort_values(key=lambda s: s.abs(), ascending=False)
print("Top Elastic Net coefficients:")
print(non_zero_coeffs.head(20))
# horrible r^2, all coefficients zero :(
```

Verify and validate your analysis:

We found that there is a significant difference in Tau ratios in high school vs post-secondary patients. We can verify our answer through the T-test using alpha=0.05. The p-value of this test

is 0.045, thus the odds of this data being randomly sorted are ~4.5%. We reject the null hypothesis.

Interestingly, we didn't find a significant correlation while running ANOVA on the given education levels: high school, graduate (PhD/Masters), bachelors, trade/tech school, and professional. In our data, differences among all educational groups were less pronounced than the contrast between the two groups.

Cai et al. shows that education has an effect on tau accumulation but is not direct and rather moderated by amyloid status such that the effect is detectable in some subgroups but may be obscured when pooled across many categories. Hoenig et al. show that higher education is associated with greater tau "buffer" (tau pathology without neuronal dysfunction) before functional decline appears, suggesting nuanced relationships between education and tau outcomes that might not manifest evenly across all education levels.

Thus, the literature supports the plausibility that an extreme contrast (high vs low education) can show significance, while across all education categories the signal may weaken. Given this, our results are not contradictory to prior findings, but rather reflect a known difficulty in detecting subtler effects across multiple strata. As our dataset was relatively small, it was difficult to

Conclusions and Ethical Implications:

Our findings support the idea that education level is meaningfully associated with tau ratios, but the effect may be most evident in broad contrasts rather than across fine-grained educational subgroups. Ethically, this calls for careful communication of results to avoid stigma- saying "low education causes worse tau pathology" risks stigmatizing individuals and overlooks confounding factors.

Additionally, if education is consistently linked to biological or cognitive outcomes (including AD), then public health/policy efforts should focus on reducing disparities. For example, they should focus on promoting lifelong learning, access to healthcare, cognitive enrichment activities, known factors that reduce neurodegenerative risk.

Limitations and Future Work:

One limitation of our project is that we were working with a relatively small dataset, which made it harder to draw strong conclusions. With fewer samples, individual outliers or unusual cases may have had a bigger effect on the results. In the future, it would be valuable to use a larger longitudinal study, which would lessen the influence of outliers and make the results less prone to bias. Tracking tau buildup over time would also make it possible to compare more directly with what the literature shows.

Notes from Team

We found that there is a significant difference in Tau ratios in high school vs post-secondary patients. There is a high correlation between the two, with a T-test showing that the odds of this

data being randomly sorted are ~4.5%. We reject the null hypothesis. This correlation is also reported in literature.

Questions for TA

None.