Supplementary Material 1: The pseudocode of the algorithm.

```
DEFINE NORMAL_BRAIN_WEIGHT_MINIMUM = 1200
DEFINE NORMAL_BRAIN_WEIGHT_MAXIMUM = 1400
DEFINE BRAIN_WEIGHT_DECREASE_AGE_THRESHOLD = 45

FUNCTION check_brain_weight_loss(age, brain_weight)
  IF age >= BRAIN_WEIGHT_DECREASE_AGE_THRESHOLD THEN
    IF brain_weight < NORMAL_BRAIN_WEIGHT_MINIMUM THEN
      PRINT "Brain weight below normal range"
    ELSE IF brain_weight > NORMAL_BRAIN_WEIGHT_MAXIMUM THEN
      PRINT "Brain weight above normal range"
    ELSE
      PRINT "Brain weight within normal range"
    ENDIF
  ELSE
    PRINT "Brain weight within normal range"
  ENDIF
END FUNCTION

FUNCTION check_cognitive_decline(age, cognitive_functions)
  IF age >= BRAIN_WEIGHT_DECREASE_AGE_THRESHOLD THEN
    IF cognitive_functions == "decline" THEN
      PRINT "Possible decline in cognitive functions"
    ELSE
      PRINT "Cognitive functions within normal range"
    ENDIF
  ELSE
    PRINT "Cognitive functions within normal range"
  ENDIF
END FUNCTION

FUNCTION check_brain_lesions(lesions_present)
  IF lesions_present THEN
    PRINT "Presence of amyloid plaques and neurofibrillary tangles"
  ELSE
    PRINT "No known pathological lesions observed"
  ENDIF
END FUNCTION

FUNCTION check_neuritic_dystrophy(amyloid_bodies, gvd, hirano_bodies)
  IF amyloid_bodies AND gvd AND hirano_bodies THEN
    PRINT "Evidence of neuritic dystrophy observed"
  ELSE
    PRINT "No evidence of neuritic dystrophy observed"
  ENDIF
END FUNCTION

################################################################
# Example usage
age = 70
brain_weight = 1150
cognitive_functions = "decline"
lesions_present = True
amyloid_bodies = True
gvd = True
hirano_bodies = True
```

```
check_brain_weight_loss(age, brain_weight)
check_cognitive_decline(age, cognitive_functions)
check_brain_lesions(lesions_present)
check_neuritic_dystrophy(amyloid_bodies, gvd, hirano_bodies)

#############################################################
age = 30
brain_weight = 1300
cognitive_functions = "normal"
lesions_present = False
amyloid_bodies = True
gvd = False
hirano_bodies = False

check_brain_weight_loss(age, brain_weight)
check_cognitive_decline(age, cognitive_functions)
check_brain_lesions(lesions_present)
check_neuritic_dystrophy(amyloid_bodies, gvd, hirano_bodies)

#############################################################
age = 40
brain_weight = 1250
cognitive_functions = "normal"
lesions_present = False
amyloid_bodies = True
gvd = True
hirano_bodies = False

check_brain_weight_loss(age, brain_weight)
check_cognitive_decline(age, cognitive_functions)
check_brain_lesions(lesions_present)
check_neuritic_dystrophy(amyloid_bodies, gvd, hirano_bodies)

#############################################################
age = 50
brain_weight = 1200
cognitive_functions = "normal"
lesions_present = False
amyloid_bodies = True
gvd = True
hirano_bodies = True

check_brain_weight_loss(age, brain_weight)
check_cognitive_decline(age, cognitive_functions)
check_brain_lesions(lesions_present)
check_neuritic_dystrophy(amyloid_bodies, gvd, hirano_bodies)
```

Supplementary Material 2: The pseudocode of the logistic regression.

```
# Import necessary libraries
import pandas as pd
from sklearn.linear_model import LogisticRegression

# Load data
data = pd.read_csv("alzheimers_data.csv")

# Split data into features and labels
X = data.drop("alzheimer", axis=1)
y = data["alzheimer"]
```

```
# Create and train logistic regression model
model = LogisticRegression()
model.fit(X, y)

# Use model to predict likelihood of Alzheimer's disease for different ages
ages = [30, 40, 50, 60, 70]
for age in ages:
    # Create new patient data with given age and other features set to average values
    new_data = {"age": age, "brain_weight": 1300, "cognitive_functions": "normal", "lesions_present": False, "amyloid_bodies": True, "gvd": age >= 40, "hirano_bodies": True}
    # Convert to DataFrame and predict likelihood of Alzheimer's disease
    new_df = pd.DataFrame(new_data, index=[0])
    likelihood = model.predict_proba(new_df)[0][1]
    # Print result
    print(f"Likelihood of Alzheimer's disease for age {age}: {likelihood:.2%}")
```