Оюутны сурлагын гүйцэтгэлийг дадал зуршил дээр үндэслэн урьдчилан таамаглах

М.Амирлангуй(23b1num1273) А.Анударь(23b1num0603)
Т.Лхагвасүрэн(23b1num1222) Э.Доржням(22b1num0166)
Э.Түвшин-Эрдэнэ(23b1num0869)

2025-12-05

Агуулга

Энэхүү ажлаар оюутнуудын өдөр тутмын дадал зуршил, суралцах хэв маяг, амьдралын хэвшлийн талаарх өгөгдөлд үндэслэн оюутны сурлагын гүйцэтгэл (GPA)-д хэрхэн нөлөөлдгийг тооцох юм. Kaggle сайтын “Student habits vs academic performance” өгөгдлийн санг ашиглан оюутны мэдээллийг оруулахад гүйцэтгэлийг урьдчилан таамаглах боломжийг бүрдүүлнэ.

Агуулга

1. Оршил

Орчин үеийн их дээд сургуулиудад оюутнуудын сурлагын гүйцэтгэлийг зөв үнэлэх, өгөгдөлд суурилсан аргачлалаар төлөв байдал, боломжит эрсдэлийг урьдчилан таамаглах шаардлага эрчимтэй өсөж байна. Ялангуяа оюутны суралцах дадал зуршил, анхаарал төвлөрөл, сурах хугацаа, мотиваци, нойр, стресс зэрэг амьдралын хэв маягийн хүчин зүйлс нь сурлагын амжилтад шууд нөлөө үзүүлдэг боловч эдгээрийг тооцоолох системтэй арга ихэнх сургуулиудад хангалтгүй ашиглагддаг.

Сүүлийн жилүүдэд машин сургалтын ангилагч загваруудыг ашиглан боловсролын салбарт сурлагын чадамжийг урьдчилан таамаглах судалгаанууд нэмэгдсээр байна. Энэхүү загвар нь өгөгдлийн шинж чанаруудыг ашиглан GPA хэд байх эсэхийг ангилахын тулд статистик магадлалыг тооцоолдог бөгөөд хурдан, гүйцэтгэл өндөртэй байдаг.

Энэхүү судалгааны ажлаар бид Kaggle сайтын “Student Habits vs Academic Performance” нэртэй өгөгдлийн санг ашиглан оюутны дадал зуршил дээр үндэслэн сурлагын амжилтын хамаарлыг судалсан юм.

Энэхүү судалгааны үндсэн зорилго нь оюутны дадал зуршил (habits) сурлагын голч дүн (GPA)-тэй хамааралтай эсэхийг тогтооход оршино.

2. Өгөгдөл ба шинжилгээ

2.1 Ашигласан өгөгдлийн сан, хувьсагчид

Энэхүү судалгаанд Kaggle сайт дахь Aryan Kumar-ын 2025 онд оруулсан Student Habits vs Academic Performance нэртэй өгөгдлийн санг ашигласан. Энэ dataset нь оюутны сурлагын гүйцэтгэлд нөлөөлж болох амьдралын хэвшил, суралцах зан төлөв, сэтгэлзүйн байдал зэрэг хүчин зүйлсийг багтаасан.

1. **Суралцах дадал ба академик хүчин зүйлс**

1. study_hours_per_day
2. attendance_percentage
3. previous_gpa
4. exam_score
5. exam_anxiety_score
6. time_management_score
7. learning_style
8. dropout_risk
9. semester
10. access_to_tutoring

2. **Сэтгэлзүй, зан төлөв ба амьдралын хэв маяг**
    1. motivation_level
    2. stress_level
    3. social_activity
    4. mental_health_rating
    5. extracurricular_participation
    6. diet_quality
    7. exercise_frequency
    8. sleep_hours
    9. screen_time
    10. netflix_hours
    11. social_media_hours
3. **Гэр бүл, санхүү ба орчин**
    1. family_income_range
    2. part_time_job
    3. parental_support_level
    4. parental_education_level
    5. internet_quality
    6. study_environment
    7. gender
    8. age
    9. major
    10. Student_id

2.2 Өгөгдлийг унших, товч шалгах

Өгөгдлийг боловсруулах үе шат: Судалгаанд ашиглах өгөгдлийн чанарыг сайжруулах зорилгоор дараах цэвэрлэгээг хийсэн. Үүнд:

1. Тооцоололд ач холбогдолгүй Student_id болон зорилтот хувьсагчтай хэт өндөр хамааралтай previous_gpa багануудыг хассан.

2. Регрессийн шинжилгээнд алдаа үүсгэхгүйн тулд дутуу утгатай (NaN) мөрүүдийг түүврээс хассан.
3. Категори хэлбэрийн өгөгдлийг (текстэн утгууд) Label Encoding аргаар тоон хэлбэрт шилжүүлсэн.

uploaded_file нь CSV форматтай гэж үзэж, pandas DataFrame-д уншина.

```
df = pd.read_csv(uploaded_file)
```

2.3 Цэвэрлэгээ ба шинж чанар боловсруулах

```
columns_to_drop = ['student_id', 'previous_gpa']
df = df.drop(columns=[col for col in columns_to_drop if col in
 ↪  df.columns], axis=1)
```

Student_id буюу оюутны дугаар нь тооцоололд шаардлагагүй, previous_gpa буюу өмнөх голч дүнг оруулснаар тооцоололд хэт хамааралтай болох учраас шаардлагагүй гэж үзэн тухайн 2 баганыг хассан.

```
df = df.dropna()
```

Өгөгдлийн бүрэн бүтэн байдлыг хангах үүднээс дутуу утгатай (missing values) мөрүүдийг түүврээс хассан.
Энэ нь regression загвар болон correlation матрицад алдаа гарахаас сэргийлдэг.

```
encoders = {}
           categorical_cols =
↪  df.select_dtypes(include=['object', 'bool']).columns
           for col in categorical_cols:
               le = LabelEncoder()
               df[col] = le.fit_transform(df[col].astype(str))
               encoders[col] = le
```

Текст төрөл буюу ‘object’, boolean буюу ‘bool’ төрлийн өгөгдөлтэй баганууудыг сонгож ‘categorical_cols’-д авна. LabelEncoder нь Male/Female гэх мэт текст утгыг 0,1 гэсэн тоонд хөрвүүлнэ. encoders[col] = le → орчуулсан mapping-г хадгалж, дараа prediction хийхдээ эсвэл inverse_transform хийхэд ашиглаж болно.

```
if 'exam_score' not in df.columns:
                st.error("Column 'exam_score' not found in the
 ↪  dataset!")
                return None, None

            df = df.rename(columns={'exam_score':
 ↪  'Target_Exam_Score'})
            return df, encoders
```

Төслийн үндсэн зорилтот багана болох exam_score өгөгдөлд байгаа эсэхийг шалгана. Байхгүй бол алдаа мэдэгдэж, None утгыг буцаана.

Дараагийн алхам model үүсгэхэд амархан ялгах зорилгоор **exam_score**-оос **Target_Exam_Score** болгон өөрчилнө.

Цэвэрлэгдсэн, боловсруулсан DataFrame (df) болон категорийн кодлогийн толь (encoders)-ийг буцаана.

```
        except Exception as e:
            st.error(f"Error loading data: {e}")
            return None, None
```

Хэрэв try блок дотор ямар нэгэн алдаа гарвал (жишээ нь, CSV файл гэмтсэн байвал), алдааны мэдэгдлийг Streamlit-ээр харуулж, None, None-ийг буцаана.

Хэрэв хэрэглэгч файл оруулаагүй бол (хамгийн эхний if uploaded_file is not None: шалгалтаар) мөн адил None, None-ийг буцаана.

3. Судалгааны арга зүй

Өгөгдөл бүтэц нь олон төрлийн тоон болон категори шинжүүдээс бүрдэх тул бид дараах регрессийн загваруудыг ашиглан гүйцэтгэлийг нь харьцуулан шинжилсэн.

Регрессийн моделууд:

1. Linear Regression: Хамгийн энгийн шугаман загвар
2. Ridge Regression: Overfitting-ийг бууруулах L2 regularization
3. Decision Tree: Шийдвэрийн мод суурьтай загвар
4. Random Forest: Олон модны ансамбль
5. SVR: Support Vector Machine суурьтай регресс
6. K-Neighbors: Хамгийн ойрын хөршүүдийн дундаж
7. Gradient Boosting: Градиент нэмэлт ансамбль

3.1 Linear Regression (Хамгийн шугаман загвар)

Шугаман регресс нь өгөгдлийн олон шинжүүд болон зорилтот хувьсагчийн хоорондын шугаман хамаарлыг илэрхийлэх үндсэн арга юм.

Ерөнхий хэлбэр:

$$\hat{y} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p$$

Энд:

- $\hat{y}$ – Таамаглагдсан шалгалтын оноо
- x_i – Оюутны зан үйл ба суралцах дадлын шинжүүд
- β_i – Тухайн шинжийн жин ба чухал байдал
- β_0 – Чөлөөт гишүүн

Шугаман регресс нь хурдтай, тохируулахад хялбар, олон шинжтэй өгөгдөлд тогтвортой тул энэ төсөлд суурь baseline болгон ашигласан.

3.2 Ridge Regression

Ridge Regression нь шугаман регрессийн өргөтгөл бөгөөд коэффициентуудын хэмжээг хэт их өсөхөөс хамгаалах L2-regularization ашигладаг.

Ерөнхий хэлбэр:

$$\min \sum (y_i - \hat{y}_i)^2 + \lambda \sum \beta_j^2$$

Энд:

- y_i : Бодит шалгалтын оноо
- $\hat{y}_i$: Таамагласан шалгалтын оноо
- β_j : Моделийн коэффициентүүд
- λ : Regularization-ийн хүчийг удирдах гиперпараметр
- $\sum(y_i - \hat{y}_i)^2$: MSE буюу загварын тайлбарлах чадвар
- $\lambda \sum \beta_j^2$: L2 шийтгэлийн гишүүн — коэффициентуудыг хэт өсөхөөс хамгаална

Энэ төсөлд Ridge() ангийг ашигласан бөгөөд feature хооронд хэт их хамаарал байж болзошгүй нөхцөлд шугаман регресстэй харьцуулахад илүү тогтвортой коэффициент өгч, ерөнхийлөх чадварыг сайжруулах зорилгоор туршсан.

3.3 Decision Tree Regressor (шийдвэрийн мод)

Decision Tree Regressor нь өгөгдлийг дараалсан шийдвэрийн цэгүүдээр (splits) хувааж, шугаман бус хамаарлыг барьж чаддаг регрессийн загвар юм.

Алгоритмын гол зарчим

- Оролтын шинжүүдээс MSE эсвэл variance-ийг хамгийн их бууруулж чадах шинжийг сонгон өгөгдлийг хуваана.
- Хуваасан хэсэг бүр дээр дахин хамгийн сайн шинжийг сонгон салбарлана.
- Эцсийн навч (leaf) node дээр тухайн хэсэгт хамаарах оноонуудын дундажийг таамаглал болгон авна.

model_builder.py дотор DecisionTreeRegressor(random_state=42) загварыг ашиглаж, оюутнуудын дадал, оролцооны түвшин, өмнөх дүнгийн янз бүрийн хослолыг шийдвэрийн модоор салбарлуулж, шугаман бус хамаарлыг илүү сайн тайлбарлаж чадах эсэхийг шалгасан.

3.4 Random Forest

Random Forest Regressor нь олон шийдвэрийн модны ансамбль загвар бөгөөд overfitting–ийг бууруулж, тогтвортой таамаглал гаргах давуу талтай.

- Сургалтын өгөгдлөөс санамсаргүй олон bootstrap дэд цуглуулга үүсгэнэ.
- Цуглуулга бүр дээр нэг шийдвэрийн мод сургаж, node бүр дээр шинжүүдийн санамсаргүй дэд хэсгийг ашиглан салбарлалт хийдэг.
- Таамаглахдаа бүх модны гаргасан $\hat{y}$ утгуудын дунджийг авна.

Давуу тал

- Overfitting багатай
- Шугаман бус хамаарлыг сайн барьдаг
- Гүйцэтгэл тогтвортой, найдвартай
- Feature importance гаргаж өгдөг

RandomForestRegressor(n_estimators=100, random_state=42) загварыг ашиглаж 100 шийдвэрийн модоор оюутны дадал, оролцоо, амьдралын хэв маяг зэрэг шинжүүдийг анализ хийж, шалгалтын оноог илүү найдвартай таамаглах боломжийг үнэлсэн. R^2 болон MAE үзүүлэлтээр бусад загвартай харьцуулж, ансамбль арга хэр сайн ажиллаж буйг шалгасан.

3.5 Support Vector Regressor – SVR)

SVR нь Support Vector Machine-ийн регрессийн хувилбар бөгөөд зорилго нь:

- Загвар оролтын өгөгдлийг өндөр хэмжээст орон зай руу kernel функцээр шилжүүлж, шугаман бус хамаарлыг илүү сайн барьдаг.

Ерөнхий хэлбэр:

$$f(x) = w^T\phi(x) + b$$

- φ(x) — kernel функцээр үүссэн шинэ онцлогийн орон зай
- w, b — загварын параметрүүд
- ε — зөвшөөрөгдөх алдааны хэмжээ

SVR нь шугаман бус, төвөгтэй хамаарлыг kernel ашиглан илүү уян хатан загварчилж чаддаг. Энэхүү төсөлд SVR() загварыг ашиглаж, оюутнуудын суралцах дадал, амьдралын хэв маяг, сэтгэл зүйн үзүүлэлтүүдийн
шугаман бус нөлөөллийг барьж, Target_Exam_Score–ийн таамаглалыг бусад загваруудтай R^2 болон MAE үзүүлэлтээр харьцуулан үнэлсэн.

3.6 K-Neighbors Regressor (Хамгийн ойрын хөршүүдийн дундаж)

K-Neighbors Regressor нь “ижил төстэй оюутнууд ижил төстэй оноо авдаг” гэсэн зарчимд тулгуурласан, энгийн бөгөөд интуитив регрессийн арга юм. Шинэ оюутны таамаглалыг хамгийн ойр K хөршийн дундаж оноогоор тооцдог.

Алгоритмын үндсэн алхам

1. Шинэ оюутны шинжийг (x⃗ₑw) өгөгдлийн орон зайд байрлуулна.
2. Өгөгдлийн багцаас түүнтэй хамгийн ойр K хөршийг
   - Евклидийн зай,
   - эсвэл бусад зайны хэмжүүрээр сонгоно.
3. Эдгээр K хөршийн бодит онооны дундажийг авч таамагласан утга (ŷ) болгон гаргана.

Ерөнхий хэлбэр:

$$\hat{y} = \frac{1}{K}\sum y_i$$

Энэ төслийн хувьд:

KNeighborsRegressor(n_neighbors=5) загварыг ашиглаж: 5 хамгийн ойр оюутны оноог дундажлан, шинэ оюутны Target_Exam_Score–ийг таамагласан. Энэ арга нь хялбардуу бөгөөд шугаман бус хамаарлыг тодорхой хэмжээнд тусгаж чаддаг тул бусад загвартай (Linear, Random Forest, SVR) хамт харьцуулалтын нэг хэсэг болгон ашигласан.

3.7 Gradient Boosting: Градиент нэмэлт ансамбль

Gradient Boosting нь олон сул регрессийн модыг дараалан сургаж, өмнөх модны гаргасан алдааг нөхөн засах замаар нэг хүчтэй ансамбль загвар бий болгодог арга юм.

Ерөнхий хэлбэр:

$$F_m(x) = F_{m-1}(x) + \nu \cdot h_m(x)$$

Энд:

- $F_m(x)$ — m-р шатны нийлмэл модель
- $h_m(x)$ — тухайн шатанд шинээр сурсан сул мод
- ν — learning rate (алдааг хэр хурдан засахыг удирдана)

Давуу тал

- Шугаман бус хамаарлыг өндөр нарийвчлалтай барина
- Алдаа засах механизмтай тул нарийвчлал сайн
- Feature importance гаргаж өгөх боломжтой

Энэ төслийн хувьд GradientBoostingRegressor(random_state=42) загварыг ажиллуулж оюутнуудын дадал, оролцоо, амьдралын хэв маяг, сэтгэлзүйн үзүүлэлтүүдийн нийлмэл нөлөөг, шат дараалсан модоор засаж таамаг гаргах боломжийг үнэлсэн.

4. Туршилт ба үр дүн

Бид цэвэрлэсэн 80000 мөр бүхий өгөгдлөөс загвар сургах хугацаа хэмнэх үүднээс санамсаргүй байдлаар 10000-ийг сонгон ашиглан linear regression, random forest, decision tree, gradient boosting, SVR загваруудыг сурган ажиллуулав. Эхлээд шалгалтын дүнг нарийвчлан таамаглах ба дараа нь практиктай нийцүүлэн 60-аас доош бол унасан, хэрэв 60-аас их буюу тэнцүү бол тэнцсэн хэмээн ангилан тохирох хэмжүүрүүдийг харьцуулав.

4.1 Регрессийн хэмжүүрүүдийн харьцуулалт

Загвар	R^2 score	MAE(mean absolute error)
Linear regression	0.167	8.79
Random forest	0.152	8.94
Decision tree	-0.755	11.87
Gradient boosting	0.181	8.69
SVR	0.013	8.93

Үүнээс дүгнэхэд gradient boosting хамгийн гүйцэтгэл сайтай байсан бол decision tree хамгийн үр дүн багатай загвар байв. Мөн linear regression random forest-оос илүү гүйцэтгэлтэй байгаа нь хүчин зүйлс болон шалгалтын оноо дийлэнхдээ шугаман хамааралтайг харуулж байж болох юм.

Мөн энд R^2 оноо бага байгаа нь манай өгөгдөл хэдий шалгалтын оноотой тодорхой хэмжээнд холбоотой боловч гадаад хүчин зүйлс ч гэсэн нөлөөлж байна гэдгийг харуулж байна.

MAE нь шалгалтын нийт оноо 100-тай харьцуулахад шалгалтандаа хангалттай авах оюутан ба хангалтгүй авах оюутныг ялган салгахад хангалттай бага байна.

4.2 Регрессийн хэмжүүрүүдийн харьцуулалт

загвар	accuracy	precision	recall	F1 score
Linear regression	98.50%	0.98	1.00	0.99
Random forest	98.50%	0.98	1.00	0.99
Decision tree	96.85%	0.99	0.98	0.98
Gradient boosting	98.50%	0.98	1.00	0.99
SVR	98.50%	0.98	1.00	0.99

Энд precision, recall, f1 score загваруудын хувьд ялгаа байхгүй байгаа ба decision tree-ээс бусад нь нарийвчлал сайтай байгааг харж болно.

4.3 Feature importance

Үзүүлэлт тус бүрийн хэр чухалыг загвар болгон дээр дараах кодыг ашиглан гаргав.

Загваруудын шинж чанарын чухал байдлыг (Feature Importance) тооцож үзэхэд загвар тус бүр өөр өөр хүчин зүйлийг онцолж байв. Жишээлбэл, Linear Regression загварын хувьд ‘Tutoring’ буюу багшаас зөвлөгөө авах боломж хамгийн чухал байсан бол бусад загваруудад ‘Study Hours’ буюу суралцах цаг голлох нөлөөтэй байна. Доорх графикаас дэлгэрэнгүйг харна уу

Эндээс харахад linear regression -ий хувьд биечлэн зөвлөгөө авах боломжтой эсэх нь хамгийн чухал байгаа бол бусад загваруудын хувьд өдөрт суралцахад зарцуулах хугацаа нь хамгийн чухал байна.

4.4 Төөрөгдлийн матриц ба бодит-таамагласан утгын график

Загвар болгонд хэр зөв, нарийвчлалтай ажилласныг харахын тулд төөрөгдлийн матриц ба бодит-таамагласан утгын графикийг дүрслэв.

5. Дүгнэлт

Судалгааны үр дүн

Манай судалгаагаар оюутны өдөр тутмын дадал зуршил, суралцах хэв маяг, амьдралын хэвшил нь сурлагын гүйцэтгэлд тодорхой нөлөө үзүүлдэг болохыг тодорхойллоо. Оюутны GPA-г урьдчилан таамаглахад хэд хэдэн машин сургалтын регрессийн загваруудыг туршиж, үр дүнг харьцуулсан.

Загварын гүйцэтгэл

Gradient Boosting regression загвар нь GPA-г урьдчилан таамаглахад хамгийн сайн гүйцэтгэл үзүүлсэн бол Linear Regression болон Random Forest загварууд мөн сайн үр дүн үзүүлсэн. Decision Tree загвар сул гүйцэтгэлтэй, R^2 бага гарсан.

Гол нөлөө үзүүлдэг хүчин зүйлс

Оюутны сурлагын амжилтанд хамгийн их нөлөө үзүүлдэг хүчин зүйлс нь өдөрт суралцах цаг, биечлэн зөвлөгөө авах боломж, мотиваци ба анхаарал төвлөрөл болох нь тодорхой болсон. Загварын R^2 бага гарсан ч MAE нь бага, ангилалтын нарийвчлал өндөр тул оюутны амжилт эсвэл уналтыг ялгахад хангалттай үр дүн үзүүлсэн. Судалгааны үр дүнд загварууд нь оноог яг таг таамаглахаас илүүтэй, оюутан шалгалтад тэнцэх эсэхийг (Binary Classification) урьдчилан таамаглахад илүү өндөр нарийвчлалтай ажиллаж байгаа нь харагдлаа.

Анхаарах зүйлс

Зарим гадаад хүчин зүйлс, жишээлбэл шалгалтын нөхцөл, багшийн үнэлгээ зэрэг нь GPA-д нөлөөлж болохыг анхаарах хэрэгтэй.

Зөвлөмж

Оюутны сурлагын амжилтыг сайжруулахын тулд өдөр тутмын суралцах хугацааг нэмэгдүүлэх, биечлэн зөвлөгөө өгөх боломжийг хангах, мотиваци, стрессийг дэмжих сургалтын арга хэмжээ авах нь чухал гэж дүгнэж болно.