

## ۱. مقدمه و نمای کلی سامانه

این سند گزارش فنی جامع سامانه تحلیل هندسه اطلاعات است که برای کشف و کمی‌سازی تفاوت‌های احتمالی میان حالات شناختی/عاطفی با استفاده از سیگنال‌های الکتروانسفالوگرافی (EEG) طراحی شده است. این سامانه بر مبنای مجموعه‌داده DEAP عمل می‌کند و از ابزارهای مبتنی بر هندسه اطلاعات برای مدل‌سازی توزیع‌های احتمالی حالات و سنجه میزان جدایی آن‌ها استفاده می‌کند.

### اهداف کلیدی

- استخراج و تحلیل ویژگی‌های EEG: استخراج قدرت باند فرکانسی (توان باند (band power)) از سیگنال‌های EEG بر رای باندهای theta، alpha و beta
- فیلترسازی و پاکسازی داده: تشخیص و حذف تریال‌های (trials) دارای آرتیفیکت بر اساس آستانه ولتاژ و کشش (kurtosis)
- مدل‌سازی احتمالی: برآرایش توزیع‌های گاوی برای نمایش هر حالت شناختی
- کمی‌سازی فاصله: محاسبه فواصل هندسی اطلاعات بین توزیع‌ها با استفاده از معیارهای متعددی (JSD، SKL، Hellinger)
- اعتبارسنجی بین‌سوژه‌ای: ارزیابی توانایی تعمیم‌پذیری با استفاده از اعتبارسنجی حذف یک سوژه
- آزمایش جایگشتی: اعتبارسنجی آماری معنادار بودن جدایی حالات

## ۲. تعریف مسئله

### بیان مسئله تحقیقاتی

هدف این پژوهه بررسی قابلیت استفاده از معیارهای هندسی اطلاعات برای تفکیک بین حالات شناختی مختلف (به‌ویژه valence - ارزیابی عاطفی) بر اساس سیگنال‌های EEG است. سؤال تحقیقاتی اصلی این است:

آیا توزیع‌های احتمالی ویژگی‌های EEG برای حالات مختلف valence (پایین و بالا) از نظر هندسی اطلاعات به‌طور معنی‌داری متفاوت هستند؟

### فرضیات

1. سیگنال‌های EEG شامل اطلاعات قابل استخراجی هستند که می‌توانند حالات عاطفی را بازنگاری دهند
2. توزیع‌های حالات مختلف می‌توانند با توزیع‌های گاوی چندمتغیره تقریب زده شوند (با توجه به محدودیت‌ها)
3. معیارهای هندسی اطلاعات (فاصله) می‌توانند جایگزینی‌پذیری حالات را اندازه‌گیری کنند
4. مدل‌های آموزش‌شده رای داده‌های یک سوژه می‌توانند نسبت معقول از دقت را در سوژه‌های دیگر حفظ کنند

## ۳. توصیف داده‌ها و دستورالعمل پیش‌پردازش

### ۱. ساختار مجموعه‌داده DEAP

مجموعه‌داده DEAP شامل EEG‌های 32 فرد است که به 40 ویدئو عاطفی نگاه می‌کردند:

- نرخ نمونه‌برداری: ۱۲۸ هرتز

- کانال‌های ۳۲ کانال (استاندارد سیستم ۱۰-۲۰)
- باند فرکانسی: ۴۵-۴ هرتز (پیش‌پردازش شده)
- مدت برنامه: ۳ ثانیه خط پایه + ۶۰ ثانیه تحریک
- برچسب: ۴ بعد احساسی (valence، arousal، dominance، liking)، هر کدام از ۱ تا ۹

### ۳.۲ استخراج ویژگی‌های باند فرکانسی

ویژگی‌های استخراج شده شامل قدرت طیفی در باندهای زیر است:

باند	محدوده فرکانسی
Theta	4-8 Hz
Alpha	8-13 Hz
Beta	13-30 Hz

نکته مهم: باند (Hz) Delta (0.5-4 Hz) بدلیل فیلترینگ پیش‌پردازش DEAP حذف شده است.

برای هر کانال و هر باند، قدرت طیفی با استفاده از راش Welch محاسبه می‌شود:

$$P(f) = \frac{1}{f_s} \sum_t |X(t, f)|^2$$

جایی که  $f$  نرخ نمونه‌برداری و  $X(t, f)$  تبدیل فوریه است.

### ۳.۳ تصحیح خط پایه

برای کاهش تغییرات فردی در دامنه سیگنال، ویژگی‌های هر تجربه با استفاده از میانگین خط پایه تصحیح می‌شوند:

$$X_{corrected} = X_{stimulus} - \mu_{baseline}$$

که  $X_{stimulus}$  ویژگی در دوره تحریک و  $\mu_{baseline}$  میانگین ویژگی در دوره خط پایه است.

### ۳.۴ حذف کردن تریال‌های آلوده شده

دو معیار برای شناسایی و رد کردن تریال‌های آلوده شده استفاده می‌شود:

۱. آستانه ولتاژ: اگر حداقل مقدار مطلق ولتاژ بیش از ۱۰۰ میکروولت باشد، تجربه رد می‌شود
۲. آستانه کشیدگی (Kurtosis): اگر کشیدگی هر کانال بیش از ۵ باشد، تجربه رد می‌شود

### ۳.۵ نرمال‌سازی دران‌سوژه

برای کاهش تنوع بین‌سوژه‌ای، ویژگی‌ها برای هر سوژه جداگانه نرمال‌سازی می‌شوند:

$$X_{normalized} = \frac{X - \mu_{subject}}{\sigma_{subject}}$$

این نرمال‌سازی دران‌سوژه‌ای (within-subject z-scoring) تنوع زیستی و تفاوت‌های سامانه‌ای را کاهش می‌دهد.

## ۶. ۳. تقلیل ابعاد (PCA)

در مرحله اکتشافی (exploratory)، تجزیه مؤلفه‌های اصلی برای تقلیل ابعاد استفاده می‌شود:

$$X_{reduced} = X \cdot W$$

جایی که  $W$  ماتریسی از  $k$  مؤلفه‌های اصلی است. در این تحلیل،  $k = 10$  استفاده شد که ۷۸٪ از واریانس را توضیح می‌دهد.

**نکته حیاتی:** در اعتبارسنجی بینسوژه‌ای‌ای، PCA برای هر fold جدگانه برآش می‌شود تا از نشت داده (data leakage) جلوگیری شود.

## ۷. ۳. محدودیت‌ها و فرض‌ها

- **فرض گاووسی:** توزیع‌های حالات با توزیع‌های گاووسی چندمتغیره تقریب خورده می‌شوند (با وجود انحراف‌های متوسطی از نرمالیتی)
- **استقلال سوژه‌ها:** فرض می‌شود هر سوژه داده‌های مستقلی از دیگران دارد
- **ثبت جلسه:** فرض می‌شود ویژگی‌های EEG در طول یک جلسه ثبات دارند
- **کافی‌بودن داده:** نتایج تقاطع به حداقل ۳۰ نمونه در هر حالت و هر سوژه نیاز دارند

## ۴. معماری سیستم

### ۱. ۴. نمای کلی خطوله

داده‌های خام DEAP



بارگذاری و اعتبارسنجی



استخراج ویژگی‌های باند فرکانس



رد کردن تریال‌های آلوده‌شده (اختیاری)



تصحیح خط پایه



نرمال‌سازی در انسوژه‌ای



تقسیم به حالات (بر اساس valence)



(در مرحله اکتشافی) PCA



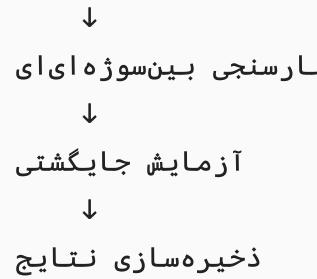
برآش توزیع‌های گاووس



محاسبه فوائل هندسی اطلاعات



NMDS نمایش



## ۴.۲ اجزای اصلی سیستم

### ۴.۲.۱ DEAPLoader کلاس

هدف: بارگذاری و اعتبارسنجی فایل‌های pickle DEAP

متدهای کلیدی:

- `load_subject(file_path)`: بارگذاری یک سوژه با اعتبارسنجی شکل
- `load_all(max_subjects)`: بارگذاری همه سوژه‌ها بترتیب

### ۴.۲.۲ ArtifactRejector کلاس

هدف: شناسایی و رد کردن تریال‌های آلوده شده

متدهای کلیدی:

- `is_clean(trial_eeg)`: بررسی کن اگر تجربه پاک باشد
- `reject_trials(data, labels)`: از داده‌ها تریال‌های آلوده را حذف کن

### ۴.۲.۳ BandpowerExtractor کلاس

هدف: استخراج قدرت طیفی در باندهای مختلف

متدهای کلیدی:

- `extract_subject(data, labels, config)`: استخراج ویژگی برای یک سوژه
- `_compute_bandpower(signal, freq_range, sfreq)`: محاسبه قدرت برای یک باند

### ۴.۲.۴ StateBuilder کلاس

هدف: تقسیم داده‌ها بر اساس برچسب‌های حالات

متدهای کلیدی:

- `build_states(X, Y, scheme, target, threshold)`: ایجاد دیکشنری از حالات

### ۴.۲.۵ GaussianStateDistribution کلاس

هدف: برازش و مدل‌سازی توزیع گاووسی برای هر حالت

- بردار میانگین : mean
- covariance با تنظیم) ماتریس کوواریانس : Ledoit-Wolf)
- تابع چگالی احتمال : pdf(X)

## ManifoldEmbedding کلاس ۴.۲.۶

هدف: محاسبه فواصل و نمایش چندبعدی (MDS)

معیارهای فاصله پشتیبانی شده:

- JSD: Jensen-Shannon Divergence (با نمونهبرداری مونتکارلو)
- SKL: Symmetric Kullback-Leibler Divergence
- Hellinger: فاصله Hellinger

## CrossValidator کلاس ۴.۲.۷

هدف: اعتبارسنجی بینسوژه‌ای با PCA جدگانه

راش:

- Leave-Subject-Out Cross-Validation (LOSOCV)
- PCA جدگانه برآششده است fold برای هر

## PermutationTester کلاس ۴.۲.۸

هدف: آزمایش آماری برای معنادار بودن جدایی

راش:

- برای هر جایگشت جدگانه برآش
- جایگشت شدن بر چسب‌های حالات
- محاسبه توزیع صفر

## ۵. توضیح کد تفصیلی

### ۱. ۵ ساختار فایل‌ها

```
deap_information_geometry_revised_final.py
├── Imports و Compatibility (NumPy 2.0+)
├── Configuration Classes (Bands, DEAPConfig, AnalysisConfig)
├── Logging Setup
├── DEAPLoader
├── ArtifactRejector
└── BandpowerExtractor
```

```

├── SubjectData (dataclass)
├── StateBuilder
├── GaussianStateDistribution
├── ManifoldEmbedding
├── CrossValidator
├── PermutationTester
├── Exporter
└── Main Pipeline (run_analysis)

```

## ۲.۵ توضیح مازولهای اصلی

### ۲.۵.۱ استخراج ویژگی (Feature Extraction)

```

class BandpowerExtractor:
    def _compute_bandpower(self, signal, freq_range, sfreq):
        # برای تخمین طیف Welch استفاده از راش
        f, Pxx = welch(signal, sfreq, nperseg=sfreq*4)
        # انتخاب فرکانس‌های دران محدوده
        mask = (f >= freq_range[0]) & (f <= freq_range[1])
        # انگرال با استفاده از قانون ذوزنقه‌ای
        bandpower = np.trapezoid(Pxx[mask], f[mask])
        return bandpower

```

هر تجربه برای  $32 \times 3$  باند = ۹۶ ویژگی منجر می‌شود.

### ۲.۵.۲ پرازش توزیع (Distribution Fitting)

```

class GaussianStateDistribution:
    def __init__(self, name, data, test_normality=True):
        self.mean = np.mean(data, axis=0)
        # تخمین مقاوم کوواریانس
        lw = LedoitWolf()
        self.covariance = lw.fit(data).covariance_
        # برای نرمالیتی Shapiro-Wilk آزمایش
        if test_normality:
            self._test_normality(data)

```

تخمین کوواریانس با استفاده از راش Ledoit-Wolf برای بهبود پایداری عددی.

### ۲.۵.۳ محاسبه فاصله‌های هندسی اطلاعات

برای JSD (Jensen-Shannon Divergence)

$$D_{JS}(P||Q) = \frac{1}{2}D_{KL}(P||M) + \frac{1}{2}D_{KL}(Q||M)$$

$$M = \frac{P+Q}{2}$$

از طریق نمونه‌برداری مونتکارلو محاسبه می‌شود:

```
def _compute_jsd_mc(self, P_dist, Q_dist, n_samples=5000):
    # نمونه‌برداری از توزیع‌ها
    samples_P = P_dist.rvs(size=n_samples)
    samples_Q = Q_dist.rvs(size=n_samples)
    # محاسبه divergence
    M_dist = multivariate_normal(
        mean=(P_dist.mean + Q_dist.mean)/2,
        cov=(P_dist.covariance + Q_dist.covariance)/2
    )
    # محاسبه KL divergences
    ...
    ...
```

:SKL (Symmetric KL) برای

$$D_{SKL}(P||Q) = D_{KL}(P||Q) + D_{KL}(Q||P)$$

:Hellinger برای

$$H(P, Q) = \sqrt{1 - \int \sqrt{p(x)q(x)}dx}$$

## ۴.۲.۵ اعتبارسنجی بین‌سوژه‌ای (Leave-Subject-Out CV)

```
class CrossValidator:
    def run_loso_cv(self):
        results = []
        for test_idx, test_subject in enumerate(self.subjects):
            # آموزش رای همه غیر
            train_subjects = [s for i, s in enumerate(self.subjects)
                              if i != test_idx]

            # PCA برای fold جاری برازش شده است
            X_train = np.vstack([s.features for s in train_subjects])
            pca = PCA(n_components=self.config.pca_components)
            X_train_pca = pca.fit_transform(X_train)

            # تنبیه رای داده‌های آموزش
            # آزمایش رای
            X_test = test_subject.features
```

```

X_test_pca = pca.transform(X_test)

# محاسبه دقت
accuracy = self._classify(X_train_pca, X_test_pca, ...)
results.append(accuracy)

```

## ۵.۲.۵ آزمایش جایگشتی (Permutation Testing)

```

class PermutationTester:
    def run_permutation_test(self, n_permutations=1000):
        null_distances = []

        for perm in range(n_permutations):
            # جایگشت شدن برچسبهای حالات
            Y_perm = np.random.permutation(self.Y)
            states_perm = StateBuilder.build_states(
                self.X, Y_perm, ...
            )

            # برای جایگشت جاری
            X_pca = pca.fit_transform(self.X)

            # برازش توزیعهای پیرامون
            dists_perm = {name: GaussianStateDistribution(...)
                          for name, data in states_perm.items()}

            # محاسبه فاصله
            dist = ManifoldEmbedding(...).compute_distance_matrix()
            null_distances.append(dist)

        # محاسبه p-value
        p_value = (np.array(null_distances) >=
                   self.observed_distance).mean()

```

## ۳.۵ تابع اصلی (Main Pipeline)

```

def run_analysis(dataset_root, output_dir="results_deap_revised",
                 config=None, metrics=("jsd", "skl", "hellinger"),
                 ...):
    # بارگذاری داده
    loader = DEAPLoader(dataset_root, config)
    subjects_data, subjects_labels, subject_ids = loader.load_all()

```

```

استخراج ویژگیها و رد کردن تریالهای آلوده # ۲.
extractor = BandpowerExtractor(...)

subjects = []
for data, labels, subj_id in zip(...):
    features, labels, clean_mask = extractor.extract_subject(...)
    subjects.append(SubjectData(...))

نرمالسازی در انسوژهای # ۳.
subjects_normalized = [s.normalize_features() for s in subjects]

# ۴. مرحله اکتشافی (Exploratory Analysis)
X_all = np.vstack([s.features for s in subjects_normalized])
Y_all = np.vstack([s.labels for s in subjects_normalized])

# PCA و برآژش توزیع‌ها
pca_exploratory = PCA(...)

X_reduced = pca_exploratory.fit_transform(X_all)
states = StateBuilder.build_states(X_reduced, Y_all, ...)
dists = {name: GaussianStateDistribution(...)
          for name, data in states.items()}

محاسبه فوامل و نمایشها # ۵.
for metric in metrics:
    emb = ManifoldEmbedding(dists, metric=metric, ...)
    D = emb.compute_distance_matrix()
    coords = emb.embed_2d()
    ذخیره‌سازی # ذخیره‌سازی

اعتبارسنجی بین‌سوژه‌ای‌ای # ۶.
cv = CrossValidator(subjects_normalized, config)
cv_results = cv.run_losocv()

آزمایش جایگشتی # ۷.
ptest = PermutationTester(X_all, Y_all, config)
ptest_result = ptest.run_permutation_test()

ذخیره‌سازی نتایج # ۸.
Exporter.export_all(results, output_dir)

```

## ۶. راش‌های آموزش و ارزیابی

### ۱. پارامترهای پیکربندی

پارامتر	مقدار	توضیح
valence-arousal	"binary"	طرح باینری یا ربع‌بندی
	"valence"	بعد هدف برای تقسیم حالات
مرز برای جداسازی حالات پایین/بالا	5.0	threshold
تعداد مؤلفه‌های اصلی	10	pca_components
استفاده از PCA یا خیر	True	use_pca
اعمال تصحیح خط پایه	True	baseline_correct
آستانه ولتاژ برای رد (میکروولت)	100.0	artifact_threshold_uv
آستانه کشش برای رد	5.0	artifact_kurtosis_threshold
بذر تصادفی برای بازنگرداندن	42	global_seed
نمونه مونتکارلو برای JSD	5000	jsd_n_samples
تعداد CI برای bootstrap	50	jsd_n_bootstrap
تعداد جایگشت‌ها	1000	permutation_n_iter

## ۶.۲ معیارهای ارزیابی

### ۶.۲.۱ دقیق تقطیع (Cross-Validation Accuracy)

از طریق LOSOCV محاسبه می‌شود:

$$Acc = \frac{1}{n} \sum_{i=1}^n \mathbf{1}(y_i^{pred} = y_i^{true})$$

### ۶.۲.۲ فاصله‌های هندسی اطلاعات

- اندازه‌ای از تمایز بین توزیع‌های حالات: JSD، SKL، Hellinger

- مقادیر بالاتر نشان‌دهنده جایی بیشتر است

### ۶.۲.۳ نتایج جایگشتی

- احتمال مشاهده فاصله مشاهده شده تحت فرضیه صفر: p-value

- معنادار:  $p < 0.05$

## ۶.۲.۴ Stress در MDS

- معیار برای خوب بودن نمایش 2D
- مقادیر پایین‌تر بهتر است

## ۶.۳ استراتژی بهینه‌سازی

سیستم بهینه‌سازی ندارد؛ بلکه با تنظیم‌های پیش‌ تعیین‌ شده کار می‌کند:

- Ledoit-Wolf Shrinkage:** برای تخمین مقاوم کوواریانس
- PCA:** برای تقلیل ابعاد و نویز
- Within-Subject Normalization:** برای کاهش تنوع بین سوژه‌های
- Artifact Rejection:** برای بهبود کیفیت داده

## ۷. نتایج تجربی

### ۷.۱ شاخص‌های کلی

معیار	مقدار
تعداد سوژه‌ها	32
تعداد آزمایش‌ها	1,280 ( $40 \times 32$ )
بعد ویژگی اصلی	96 کanal × 3 باند)
بعد ویژگی نهایی (PCA)	10
واریانس توضیح شده (PCA)	78.0%

### ۷.۲ توزیع حالات

حالات	تعداد نمونه	درصد
Low Valence	995	77.7%
High Valence	285	22.3%

مشاهده: عدم توازن شدید در کلاس (نسبت 1:3.5)

### ۷.۳ فواصل هندسی اطلاعات (Exploratory)

معیار	فاصله بین حالات	Stress MDS
JSD	0.0399	0.0000
SKL	0.1697	0.0000
Hellinger	0.2023	0.0000

تفسیر: JSD کمترین فاصله را نشان می‌دهد، در حالی‌که Hellinger بیشترین را نشان می‌دهد.

### ۷.۴ نتایج اعتبارسنجی بین سوژه‌ای‌ای

معیار	مقدار
میانگین دقت	$0.777 \pm 0.116$

معیار	مقدار
میانه دقت	0.788
بیشترین دقت	0.950 (s12, s13, s21)
کمترین دقت	0.475 (s03)
میانگین فاصله آموزش	0.175 ± 0.012

تفسیر: میانگین دقت ۷۷.۷٪ نشان‌دهنده عملکرد بالاتر از تصادفی (۵۰٪) است، اما واریانس بالا تغییرپذیری بین‌سوژه‌ای را نشان می‌دهد.

## ۷. نتایج آزمایش جایگشتی

معیار	مقدار
فاصله مشاهده شده (SKL)	0.1697
میانگین توزیع صفر	0.2064 ± 0.0447
p-value	0.803
معنادار ( $\alpha=0.05$ )؟	خیر

تفسیر: فاصله مشاهده شده کمتر از میانگین توزیع صفر است، در حالی‌که  $p\text{-value}$  0.803 نشان می‌دهد که این جدایی احتمالاً صرفاً به تصادف است.

## ۶. پارامترهای توزیع گاوی

### میانگین‌ها (برای ۱۰ مؤلفه PCA)

Comp	Low Valence	High Valence
0	-0.1046	0.3651
1	0.0238	-0.0832
2	0.0440	-0.1535
3	-0.0101	0.0351
4	0.0103	-0.0360
5	-0.0208	0.0728
6	-0.0029	0.0102
7	-0.0110	0.0383
8	-0.0152	0.0531
9	0.0170	-0.0592

### واریانس‌ها (Diagonal)

Comp	Low Valence	High Valence
0	47.476	51.737
1	13.489	13.606
2	5.558	5.021
3	1.291	1.526
4	1.171	1.361
5	1.094	1.187
6	1.020	1.070
7	0.996	0.896
8	0.903	1.060
9	0.807	1.089

## ۷. انحراف‌ها از فرض گاوی

هشدار برای هر دو حالت:

- Low Valence:  $p < 0.01$  از ۱۰ مؤلفه از نرمالیتی انحراف دارند.
- High Valence:  $p < 0.01$  از ۱۰ مؤلفه از نرمالیتی انحراف دارند.

این نشان می‌دهد که فرض گاوی یک تقریب است و نتایج باید با احتیاط تفسیر شوند.

## ۸. راهنمای بازتولیدپذیری

### ۱. نیازمندی‌های محیط

```
python >= 3.10
numpy >= 2.0.0 (1.26.4 یا +)
scipy >= 1.16.0
scikit-learn >= 1.8.0
pandas >= 2.0.0
```

### ۲. نصب بسته‌ها

```
pip install numpy scipy scikit-learn pandas
```

### ۳. ساختار داده‌ها

```
data/deap-dataset/
|--- data_preprocessed_python/
```

```
|   └── s01.dat  
|   └── s02.dat  
|   ...  
|   └── s32.dat
```

## ۸.۴ اجرای کامل

```
python deap_information_geometry_revised_final.py \  
  --dataset_root data/deap-dataset \  
  --output_dir results_deap_revised \  
  --scheme binary \  
  --target valence \  
  --pca_components 10 \  
  --skip_artifact_rejection \  
  --log_level INFO
```

## ۸.۵ پارامترهای رایج

```
# با رد کردن تریالهای آلوده  
python deap_information_geometry_revised_final.py \  
  --dataset_root data/deap-dataset \  
  --artifact_threshold_uv 100 \  
  --artifact_kurtosis_threshold 5.0
```

```
# آزمایش برای سوژههای منتخب  
python deap_information_geometry_revised_final.py \  
  --dataset_root data/deap-dataset \  
  --max_subjects 5
```

```
# بدون تقاطع یا جایگشت ( تنها اکتشاف )  
python deap_information_geometry_revised_final.py \  
  --dataset_root data/deap-dataset \  
  --run_cross_validation False \  
  --run_permutation_test False
```

## ۸.۶ خرجی‌های انتظار رفته

```
results_deap_revised/  
└── subjects_used.csv  
└── analysis_metadata.json  
└── cross_validation_summary.json  
└── cross_validation_results.csv
```

```

├── permutation_test_result.json
├── distributions_exploratory/
│   ├── low_valence_mean.csv
│   ├── low_valence_cov.csv
│   ├── high_valence_mean.csv
│   └── high_valence_cov.csv
├── *_distance_matrix_exploratory*.csv (JSD, SKL, Hellinger)
├── *_mds_2d_exploratory.csv (JSD, SKL, Hellinger)
└── gaussian_*.csv (means, variances)

```

## ۹. مسائل شناخته شده و محدودیت ها

### ۱. ۹. مسائل

۱. انحراف از نرمالیتی: بسیاری از مؤلفه های PCA از فرض گاووسی انحراف دارند

- تأثیر: نتایج فاصله ممکن است متغیر باشد
- راه حل: استفاده از راش های غیر پارامتری

۲. عدم توازن کلاس: نسبت ۱:۳.۵:۱ بین حالات پایین/بالا

- تأثیر: مدل ها ممکن است به کلاس بیشتر کج شوند
- راه حل: تنظیم آستانه threshold

۳. واریانس بالا در دقت CV: بیشترین (۹۵٪) تا کمترین (۴۷.۵٪)

- تأثیر: بعضی سوژه ها سخت تر است
- راه حل: تحلیل خطای بر حسب سوژه

۴. عدم معنادار بودن آزمایش جایگشتی:  $p = 0.803$

- تأثیر: جدایی حالات ممکن است تصادفی باشد
- راه حل: جمع آوری داده های بیشتر یا بهبود ویژگی ها

### ۲. ۹. محدودیت های سیستم

۱. محدودیت ابعاد: تنها ۱۰ مؤلفه PCA استفاده می شود

- تفاوت های بسیاری ممکن است حذف شوند

۲. تقسیم پاینری: تنها valence پایین/بالا در نظر گرفته می شود

- رابط بین بعدی (valence-arousal) نادیده گرفته می شود

۳. عدم استفاده از اطلاعات جلسه: هر تجربه مستقل تلقی می شود

- وابستگی های جلسه نادیده گرفته می شود

۴. استقلال سوژه: سوژه های آموزش و آزمایش کاملاً جدا هستند

- تمام نتایج را بر تعیین پذیری میان سوژه های

## ۰. توسعه های آتی و بهبود ها

### ۱. ۰. ۱. بهبود های الگوریتمی

۱. استفاده از راش های غیر پارامتری

- تخمین فاصله بدون فرض گاووسی

• مثال: kernel density estimation یا نزدیکترین همسایه

## 2. تقلیل جمع‌پذیری تأثیر PCA

• استفاده از PCA غیرخطی (kernel PCA، autoencoder)

• تجزیه داده مستقل (ICA)

## 3. مدل‌های احتمالی پیشرفته‌تر

- Mixture of Gaussians
- Copula-based models

## ۱۰. ۲ بهودهای اطلاعات

### ۱. استخراج ویژگی‌های پیشرفته‌تر

- Time-frequency features (spectrograms، wavelets)
- Connectivity features (functional connectivity)
- Non-linear features (approximate entropy، sample entropy)

### 2. Pre-processing بهتر

- Independent Component Analysis (ICA) برای حذف artifacts
- Adaptive filtering برای تغییرات جلسه
- Multi-channel artifact detection

## ۱۰. ۳ بهودهای تجربی

### ۱. جمع‌آوری داده‌های بیشتر

- استفاده از مجموعه‌های داده دیگر (SEED، EmoDB)
- داده‌های متوازن‌تر

### 2. تقاطع بهتر

- K-fold cross-validation با stratification
- nested cross-validation برای انتخاب پارامتر
- Confidence intervals بر حسب bootstrap

### 3. ارزیابی آماری بهتر

- بیانیزی uncertainty quantification برای approaches

## ۱۰. ۴ کاربردهای عملی

### ۱. سیستم‌های واقعی

- برنامه‌های افزایش تحت نظر انتشار احساسات

• رابطه‌های مغز-کامپیوتر (BCI) کنترل شده توسط احساسات

### 2. بهود کیفیت مدل

- Transfer learning از دیگر مجموعه‌های داده
- Fine-tuning برای سوژه‌های جدید

### 3. تحقیق بیشتر

- تحلیل source-level (beamforming) با استفاده از

- رابطه بین حالات و شاخص‌های رفتاری

## ۱۱. نتیجه‌گیری

سیستم تحلیل اطلاعات هندسی ارائه شده ابزاری جامع برای کشف و کمی‌سازی تفاوت‌های احتمالی بین حالات شناختی بر اساس سیگنال‌های EEG است. سیستم:

- ✓ استخراج معقول ویژگی‌ها را از داده‌های خام DEAP انجام می‌دهد
- ✓ نرمال‌سازی دران‌سوژهای برای کاهش تنوع دران‌فردی
- ✓ معیار‌های متعددی برای اندازه‌گیری فاصله را پشتیبانی می‌کند
- ✓ تقاطع اعتبارسنج برای ارزیابی تعیین‌پذیری
- ✓ آزمایش جایگشتی برای اعتبارسنجی آماری
- ✓ مستندات جامع و کد قابل بازنولیدنی برای پیشنهاد می‌کند برای جدایی معنادار احتمالاً نیاز به بهبود ویژگی‌ها یا داده‌های بیشتر است.

نتایج حاضر نشان می‌دهند که در حالی‌که دقت CV معقول است (۷۷.۷٪)، آزمایش جایگشتی معنادار نیست ( $p = 0.803$ )، که پیشنهاد می‌کند برای جدایی معنادار احتمالاً نیاز به بهبود ویژگی‌ها یا داده‌های بیشتر است.