**📘 Ausführliche Tabellenbeschreibung**

1. **frzk\_operatoren**
   * Zweck: Abbildung der vier didaktischen Operatoren σ, M, R, E als empirische Messgrößen.
   * Bezug FRZK: Direkt aus Kap. 6.2, dort als Operatorenkaskade beschrieben.
   * Nutzen: Erlaubt quantitative und qualitative Analyse, wie stark unterschiedliche Operatoren in Lernprozessen auftreten.
2. **frzk\_loops**
   * Zweck: Erfasst Phasen von Verdichtung (hohe semantische Aktivität) und Pausen (Stillstand/Instabilität).
   * Bezug FRZK: Kapitel 6.2.4–6.2.6 (Loop-Struktur als Lernrhythmus).
   * Nutzen: Analysen von Lernrhythmen, Stabilität/Instabilität, Identifikation von optimalen Pausen- und Resonanzphasen.
3. **frzk\_transitions**
   * Zweck: Dokumentiert Übergänge/Kippmomente, wenn neue Strukturen oder Bedeutungen emergieren.
   * Bezug FRZK: Kapitel 3.10 und 6.3 (Emergenz, σ-Zonen, Dichte-Übergänge).
   * Nutzen: Empirische Prüfung der These, dass Lernen als Sequenz von Übergängen/Kippmomenten verläuft.
4. **frzk\_reflexion**
   * Zweck: Trennung von Beobachterebenen (Selbst, Gruppe, Lehrkraft, Forscher).
   * Bezug FRZK: Kapitel 3.9 und 6.2.7 (Beobachtung/Zuweisung, Metaebene).
   * Nutzen: Erlaubt, Selbstberichte der Lernenden, Lehrkraftkommentare und Forscherkodierungen systematisch zu vergleichen.
5. **frzk\_interdependenz**
   * Zweck: Statistische Beschreibung des Zusammenspiels zwischen kognitiver, sozialer, affektiver und semantischer Dimension.
   * Bezug FRZK: Kapitel 3.4–3.5 (Raum als Zuweisung, höhere Dimensionen).
   * Nutzen: Liefert quantitative Evidenz, wie stark die Dimensionen gekoppelt sind, ob Balance oder Dominanz einzelner Achsen vorliegt.

Um die neuen **FRZK-Erweiterungstabellen** sinnvoll zu befüllen, musst du verschiedene Datenquellen aus deiner Feldstudie (icas.sql) nutzen und mit Auswertungslogik kombinieren. Ich gebe dir eine **praktische Roadmap pro Tabelle**, welche Daten du woher ziehst und wie du sie einspeist:

**1. frzk\_operatoren**

**Zweck:** Abbildung der Operatoren σ, M, R, E.  
**Befüllung:**

* Quelle: mtr\_didaktik, mtr\_rueckkopplung\_lehrkraft\_\*, mtr\_leistung.
* Mapping:
  + sigma\_level → Themenauswahl, Klarheit von Aufgaben (Semantisierung).
  + m\_level → Reflexionshinweise, Meta-Kommentare (Meta-Operator).
  + r\_level → Lehrkraft-Rückkopplung, Resonanz-Scores.
  + e\_level → Auftreten neuer Cluster in frzk\_semantische\_dichte.
* Befüllung: SQL-INSERT nach Berechnung/Skalierung (z. B. 0–5).

**2. frzk\_loops**

**Zweck:** Lernrhythmen, Verdichtung & Pause.  
**Befüllung:**

* Quelle: frzk\_semantische\_dichte.
* Logik:
  + startzeit = erster Eintrag mit steigender σ.
  + endzeit = Zeitpunkt, an dem σ-Wert wieder stabilisiert oder aufhört.
  + verdichtungsgrad = max σ – min σ innerhalb des Zeitfensters.
  + pausenmarker = TRUE, wenn dh/dt ≈ 0 über längeren Zeitraum.

**3. frzk\_transitions**

**Zweck:** Kippmomente, Emergenz.  
**Befüllung:**

* Quelle: frzk\_semantische\_dichte (dh/dt, stabilitaet\_score) + mtr\_emotions.
* Logik:
  + Ein transition wird erzeugt, wenn:
    - σ-Sprung über Schwellwert (z. B. Δσ > 0.5).
    - Oder plötzlicher Emotionswechsel (Freude → Frustration).
  + typ setzt du nach Heuristik: Stabilisierung, Irritation, Bedeutungswechsel.

**4. frzk\_reflexion**

**Zweck:** Beobachterebenen trennen.  
**Befüllung:**

* Quelle:
  + Selbstberichte (Fragebogen/Reflexionen der TN).
  + Lehrkraft-Notizen (z. B. aus mtr\_rueckkopplung\_lehrkraft).
  + Forscherkodierungen (aus Auswertung).
* Vorgehen:
  + Manuell oder halbautomatisch einfügen.
  + ebene je nach Quelle (Selbst/Gruppe/Lehrkraft/Forscher).

**5. frzk\_interdependenz**

**Zweck:** Zusammenspiel der Dimensionen.  
**Befüllung:**

* Quelle: Kombination aus
  + mtr\_leistung (kognitiv),
  + map\_tmp\_ue (sozial),
  + mtr\_emotions (affektiv),
  + frzk\_semantische\_dichte.h\_bedeutung.
* Logik:
  + Aggregation pro Zeitfenster (Mittelwerte).
  + Berechnung Korrelation/Regression (z. B. SPSS, R oder SQL-Funktion).
  + Ergebnisse als korrelationsscore speichern.

**🚀 Praktisches Vorgehen**

1. **Datenextraktion:** Mit SQL-Views oder Python/R-Skripten die relevanten Variablen aus den bestehenden Tabellen ziehen.
2. **Transformation:** Berechnungen durchführen (σ-Sprünge, Stabilität, Operatorwerte).
3. **Befüllung:** Ergebnisse in die neuen Tabellen schreiben (INSERT INTO …).
4. **Validierung:** Testen, ob die Werte plausible Muster ergeben (z. B. Loops zeigen Verdichtung & Pausen).

<?php

// ================================================

// FRZK-Erweiterung: Automatische Befüllung neuer Tabellen aus icas.sql

// Autor: (dein Name)

// ================================================

// 1. PDO-Verbindung

$dsn = "mysql:host=localhost;dbname=icas;charset=utf8mb4";

$user = "root";

$pass = "";

try {

$pdo = new PDO($dsn, $user, $pass, [

PDO::ATTR\_ERRMODE => PDO::ERRMODE\_EXCEPTION

]);

} catch (PDOException $e) {

die("DB-Verbindung fehlgeschlagen: " . $e->getMessage());

}

// ===================================================

// 2. Befüllung frzk\_loops (Verdichtung & Pause)

// ===================================================

$query = "SELECT teilnehmer\_id, zeitpunkt, h\_bedeutung, dh\_dt, stabilitaet\_score

FROM frzk\_semantische\_dichte

ORDER BY teilnehmer\_id, zeitpunkt";

$stmt = $pdo->query($query);

$currentLoop = null;

foreach ($stmt as $row) {

$tid = $row['teilnehmer\_id'];

$zeit = $row['zeitpunkt'];

$sigma = $row['h\_bedeutung'];

$delta = $row['dh\_dt'];

$stab = $row['stabilitaet\_score'];

// Loop starten, wenn σ steigt

if ($currentLoop === null && $delta > 0.1) {

$currentLoop = [

'teilnehmer\_id' => $tid,

'startzeit' => $zeit,

'sigma\_max' => $sigma,

'sigma\_min' => $sigma

];

}

// Loop läuft -> max/min aktualisieren

if ($currentLoop !== null) {

$currentLoop['sigma\_max'] = max($currentLoop['sigma\_max'], $sigma);

$currentLoop['sigma\_min'] = min($currentLoop['sigma\_min'], $sigma);

// Loop beenden, wenn Stabilität hoch oder dh\_dt ~ 0

if ($stab > 0.8 || abs($delta) < 0.05) {

$verdichtungsgrad = $currentLoop['sigma\_max'] - $currentLoop['sigma\_min'];

$pausenmarker = ($stab > 0.9) ? 1 : 0;

$insert = $pdo->prepare("

INSERT INTO frzk\_loops

(teilnehmer\_id, startzeit, endzeit, verdichtungsgrad, pausenmarker)

VALUES (:tid, :start, :end, :verd, :pause)

");

$insert->execute([

':tid' => $currentLoop['teilnehmer\_id'],

':start' => $currentLoop['startzeit'],

':end' => $zeit,

':verd' => $verdichtungsgrad,

':pause' => $pausenmarker

]);

$currentLoop = null; // Reset

}

}

}

// ===================================================

// 3. Befüllung frzk\_operatoren

// ===================================================

$query2 = "SELECT id, teilnehmer\_id, datum, lernfortschritt, transferdenken, reflexionshinweis, beherrscht\_thema

FROM mtr\_leistung";

$stmt2 = $pdo->query($query2);

foreach ($stmt2 as $row) {

$sigma = $row['lernfortschritt']; // σ ~ Semantisierung

$m = $row['transferdenken']; // M ~ Meta

$r = $row['reflexionshinweis']; // R ~ Resonanz

$e = $row['beherrscht\_thema']; // E ~ Emergenz

$insert2 = $pdo->prepare("

INSERT INTO frzk\_operatoren

(teilnehmer\_id, zeitpunkt, sigma\_level, m\_level, r\_level, e\_level, bemerkung)

VALUES (:tid, :zeit, :sigma, :m, :r, :e, :bem)

");

$insert2->execute([

':tid' => $row['teilnehmer\_id'],

':zeit' => $row['datum'],

':sigma' => $sigma,

':m' => $m,

':r' => $r,

':e' => $e,

':bem' => 'Automatisch importiert aus mtr\_leistung'

]);

}

// ===================================================

// 4. Befüllung frzk\_transitions (Kippmomente)

// ===================================================

$query3 = "SELECT teilnehmer\_id, zeitpunkt, h\_bedeutung, dh\_dt, stabilitaet\_score

FROM frzk\_semantische\_dichte

ORDER BY teilnehmer\_id, zeitpunkt";

$stmt3 = $pdo->query($query3);

$last = [];

foreach ($stmt3 as $row) {

$tid = $row['teilnehmer\_id'];

$zeit = $row['zeitpunkt'];

$sigma = $row['h\_bedeutung'];

if (isset($last[$tid])) {

$deltaSigma = abs($sigma - $last[$tid]['sigma']);

if ($deltaSigma > 0.5) { // Schwellwert für Transition

$insert3 = $pdo->prepare("

INSERT INTO frzk\_transitions

(teilnehmer\_id, zeitpunkt, typ, indikator\_score, kommentar)

VALUES (:tid, :zeit, :typ, :score, :komm)

");

$insert3->execute([

':tid' => $tid,

':zeit' => $zeit,

':typ' => 'Bedeutungswechsel',

':score' => $deltaSigma,

':komm' => 'Δσ > 0.5 erkannt'

]);

}

}

$last[$tid] = ['sigma' => $sigma];

}

// ===================================================

// 5. Befüllung frzk\_reflexion (Meta-Ebene)

// ===================================================

// Hier ein Dummy-Import: z. B. Lehrkraft-Rückmeldungen als Ebene "Lehrkraft"

$query4 = "SELECT id, teilnehmer\_id, datum, rueckmeldung

FROM mtr\_rueckkopplung\_lehrkraft\_lesson";

$stmt4 = $pdo->query($query4);

foreach ($stmt4 as $row) {

$insert4 = $pdo->prepare("

INSERT INTO frzk\_reflexion

(teilnehmer\_id, beobachter\_id, ebene, datum, reflexionstext)

VALUES (:tid, :bid, :ebene, :datum, :text)

");

$insert4->execute([

':tid' => $row['teilnehmer\_id'],

':bid' => 1, // ID der Lehrkraft (Dummy, anpassen)

':ebene' => 'Lehrkraft',

':datum' => $row['datum'],

':text' => $row['rueckmeldung']

]);

}

// ===================================================

// 6. Befüllung frzk\_interdependenz (Korrelationen)

// ===================================================

// Vereinfachtes Beispiel: Mittelwerte pro Teilnehmer berechnen

$query5 = "SELECT l.teilnehmer\_id,

AVG(l.lernfortschritt) AS avg\_x,

AVG(e.interesse) AS avg\_z,

AVG(d.methodenvielfalt) AS avg\_y

FROM mtr\_leistung l

JOIN mtr\_emotions e ON l.teilnehmer\_id = e.teilnehmer\_id

JOIN mtr\_didaktik d ON l.teilnehmer\_id = d.teilnehmer\_id

GROUP BY l.teilnehmer\_id";

$stmt5 = $pdo->query($query5);

foreach ($stmt5 as $row) {

$score = ($row['avg\_x'] + $row['avg\_y'] + $row['avg\_z']) / 3; // Dummy-Korrelation

$insert5 = $pdo->prepare("

INSERT INTO frzk\_interdependenz

(zeitpunkt, x\_kognition, y\_sozial, z\_affektiv, h\_bedeutung, korrelationsscore, methode)

VALUES (NOW(), :x, :y, :z, :h, :score, :meth)

");

$insert5->execute([

':x' => $row['avg\_x'],

':y' => $row['avg\_y'],

':z' => $row['avg\_z'],

':h' => $score,

':score' => $score,

':meth' => 'Mittelwertdummy (noch ersetzen durch echte Korrelationsanalyse)'

]);

}

echo "Alle FRZK-Tabellen erfolgreich befüllt!\n";

<?php

// ================================================

// FRZK-Erweiterung: Automatische Befüllung neuer Tabellen aus icas.sql

// Autor: (dein Name)

// ================================================

// 1. PDO-Verbindung

$dsn = "mysql:host=localhost;dbname=icas;charset=utf8mb4";

$user = "root";

$pass = "";

try {

$pdo = new PDO($dsn, $user, $pass, [

PDO::ATTR\_ERRMODE => PDO::ERRMODE\_EXCEPTION

]);

} catch (PDOException $e) {

die("DB-Verbindung fehlgeschlagen: " . $e->getMessage());

}

// ===================================================

// 2. Befüllung frzk\_loops (Verdichtung & Pause)

// ===================================================

$query = "SELECT teilnehmer\_id, zeitpunkt, h\_bedeutung, dh\_dt, stabilitaet\_score

FROM frzk\_semantische\_dichte

ORDER BY teilnehmer\_id, zeitpunkt";

$stmt = $pdo->query($query);

$currentLoop = null;

foreach ($stmt as $row) {

$tid = $row['teilnehmer\_id'];

$zeit = $row['zeitpunkt'];

$sigma = $row['h\_bedeutung'];

$delta = $row['dh\_dt'];

$stab = $row['stabilitaet\_score'];

// Loop starten, wenn σ steigt

if ($currentLoop === null && $delta > 0.1) {

$currentLoop = [

'teilnehmer\_id' => $tid,

'startzeit' => $zeit,

'sigma\_max' => $sigma,

'sigma\_min' => $sigma

];

}

// Loop läuft -> max/min aktualisieren

if ($currentLoop !== null) {

$currentLoop['sigma\_max'] = max($currentLoop['sigma\_max'], $sigma);

$currentLoop['sigma\_min'] = min($currentLoop['sigma\_min'], $sigma);

// Loop beenden, wenn Stabilität hoch oder dh\_dt ~ 0

if ($stab > 0.8 || abs($delta) < 0.05) {

$verdichtungsgrad = $currentLoop['sigma\_max'] - $currentLoop['sigma\_min'];

$pausenmarker = ($stab > 0.9) ? 1 : 0;

$insert = $pdo->prepare("

INSERT INTO frzk\_loops

(teilnehmer\_id, startzeit, endzeit, verdichtungsgrad, pausenmarker)

VALUES (:tid, :start, :end, :verd, :pause)

");

$insert->execute([

':tid' => $currentLoop['teilnehmer\_id'],

':start' => $currentLoop['startzeit'],

':end' => $zeit,

':verd' => $verdichtungsgrad,

':pause' => $pausenmarker

]);

$currentLoop = null; // Reset

}

}

}

// ===================================================

// 3. Befüllung frzk\_operatoren

// ===================================================

$query2 = "SELECT id, teilnehmer\_id, datum, lernfortschritt, transferdenken, reflexionshinweis, beherrscht\_thema

FROM mtr\_leistung";

$stmt2 = $pdo->query($query2);

foreach ($stmt2 as $row) {

$sigma = $row['lernfortschritt']; // σ ~ Semantisierung

$m = $row['transferdenken']; // M ~ Meta

$r = $row['reflexionshinweis']; // R ~ Resonanz

$e = $row['beherrscht\_thema']; // E ~ Emergenz

$insert2 = $pdo->prepare("

INSERT INTO frzk\_operatoren

(teilnehmer\_id, zeitpunkt, sigma\_level, m\_level, r\_level, e\_level, bemerkung)

VALUES (:tid, :zeit, :sigma, :m, :r, :e, :bem)

");

$insert2->execute([

':tid' => $row['teilnehmer\_id'],

':zeit' => $row['datum'],

':sigma' => $sigma,

':m' => $m,

':r' => $r,

':e' => $e,

':bem' => 'Automatisch importiert aus mtr\_leistung'

]);

}

// ===================================================

// 4. Befüllung frzk\_transitions (Kippmomente)

// ===================================================

$query3 = "SELECT teilnehmer\_id, zeitpunkt, h\_bedeutung, dh\_dt, stabilitaet\_score

FROM frzk\_semantische\_dichte

ORDER BY teilnehmer\_id, zeitpunkt";

$stmt3 = $pdo->query($query3);

$last = [];

foreach ($stmt3 as $row) {

$tid = $row['teilnehmer\_id'];

$zeit = $row['zeitpunkt'];

$sigma = $row['h\_bedeutung'];

if (isset($last[$tid])) {

$deltaSigma = abs($sigma - $last[$tid]['sigma']);

if ($deltaSigma > 0.5) { // Schwellwert für Transition

$insert3 = $pdo->prepare("

INSERT INTO frzk\_transitions

(teilnehmer\_id, zeitpunkt, typ, indikator\_score, kommentar)

VALUES (:tid, :zeit, :typ, :score, :komm)

");

$insert3->execute([

':tid' => $tid,

':zeit' => $zeit,

':typ' => 'Bedeutungswechsel',

':score' => $deltaSigma,

':komm' => 'Δσ > 0.5 erkannt'

]);

}

}

$last[$tid] = ['sigma' => $sigma];

}

// ===================================================

// 5. Befüllung frzk\_reflexion (Meta-Ebene)

// ===================================================

// Hier ein Dummy-Import: z. B. Lehrkraft-Rückmeldungen als Ebene "Lehrkraft"

$query4 = "SELECT id, teilnehmer\_id, datum, rueckmeldung

FROM mtr\_rueckkopplung\_lehrkraft\_lesson";

$stmt4 = $pdo->query($query4);

foreach ($stmt4 as $row) {

$insert4 = $pdo->prepare("

INSERT INTO frzk\_reflexion

(teilnehmer\_id, beobachter\_id, ebene, datum, reflexionstext)

VALUES (:tid, :bid, :ebene, :datum, :text)

");

$insert4->execute([

':tid' => $row['teilnehmer\_id'],

':bid' => 1, // ID der Lehrkraft (Dummy, anpassen)

':ebene' => 'Lehrkraft',

':datum' => $row['datum'],

':text' => $row['rueckmeldung']

]);

}

// ===================================================

// 6. Befüllung frzk\_interdependenz (Korrelationen)

// ===================================================

// Vereinfachtes Beispiel: Mittelwerte pro Teilnehmer berechnen

$query5 = "SELECT l.teilnehmer\_id,

AVG(l.lernfortschritt) AS avg\_x,

AVG(e.interesse) AS avg\_z,

AVG(d.methodenvielfalt) AS avg\_y

FROM mtr\_leistung l

JOIN mtr\_emotions e ON l.teilnehmer\_id = e.teilnehmer\_id

JOIN mtr\_didaktik d ON l.teilnehmer\_id = d.teilnehmer\_id

GROUP BY l.teilnehmer\_id";

$stmt5 = $pdo->query($query5);

foreach ($stmt5 as $row) {

$score = ($row['avg\_x'] + $row['avg\_y'] + $row['avg\_z']) / 3; // Dummy-Korrelation

$insert5 = $pdo->prepare("

INSERT INTO frzk\_interdependenz

(zeitpunkt, x\_kognition, y\_sozial, z\_affektiv, h\_bedeutung, korrelationsscore, methode)

VALUES (NOW(), :x, :y, :z, :h, :score, :meth)

");

$insert5->execute([

':x' => $row['avg\_x'],

':y' => $row['avg\_y'],

':z' => $row['avg\_z'],

':h' => $score,

':score' => $score,

':meth' => 'Mittelwertdummy (noch ersetzen durch echte Korrelationsanalyse)'

]);

}

echo "Alle FRZK-Tabellen erfolgreich befüllt!\n";

👉 Soll ich dir auch noch ein **zweites Statistikmaß** (z. B. **lineare Regression** oder **Spearman-Korrelation**) ins Skript einbauen, damit du die Interdependenzen unterschiedlich berechnen kannst?

**Du:**

ja

**ChatGPT:**

Perfekt 👍  
Dann erweitere ich das Skript um zwei zusätzliche Statistik-Maße:

1. **Spearman-Rangkorrelation** (robust bei ordinalen Skalen, nicht normalverteilten Daten)
2. **Lineare Regression** (einfache Regression X → Y, liefert Steigung & Bestimmtheitsmaß R²)

Damit kannst du frzk\_interdependenz flexibler befüllen.

**📄 frzk\_import.php (erweitert)**

<?php

// ================================================

// FRZK: Berechnung von Interdependenzen

// Pearson, Spearman, Regression

// ================================================

$dsn = "mysql:host=localhost;dbname=icas;charset=utf8mb4";

$user = "root";

$pass = "";

try {

$pdo = new PDO($dsn, $user, $pass, [

PDO::ATTR\_ERRMODE => PDO::ERRMODE\_EXCEPTION

]);

} catch (PDOException $e) {

die("DB-Verbindung fehlgeschlagen: " . $e->getMessage());

}

// ------------------------------------------------

// Pearson-Korrelation

// ------------------------------------------------

function pearsonCorrelation(array $x, array $y): ?float {

$n = count($x);

if ($n !== count($y) || $n === 0) return null;

$meanX = array\_sum($x) / $n;

$meanY = array\_sum($y) / $n;

$num = 0.0; $denX = 0.0; $denY = 0.0;

for ($i = 0; $i < $n; $i++) {

$dx = $x[$i] - $meanX;

$dy = $y[$i] - $meanY;

$num += $dx \* $dy;

$denX += $dx \* $dx;

$denY += $dy \* $dy;

}

if ($denX == 0 || $denY == 0) return null;

return $num / sqrt($denX \* $denY);

}

// ------------------------------------------------

// Spearman-Korrelation

// ------------------------------------------------

function spearmanCorrelation(array $x, array $y): ?float {

$n = count($x);

if ($n !== count($y) || $n === 0) return null;

// Ränge berechnen

$rank = function($arr) {

$sorted = $arr;

asort($sorted);

$ranks = [];

$rank = 1;

foreach ($sorted as $key => $val) {

$ranks[$key] = $rank++;

}

return $ranks;

};

$rankX = $rank($x);

$rankY = $rank($y);

// Pearson auf Rängen anwenden

$rx = array\_values($rankX);

$ry = array\_values($rankY);

return pearsonCorrelation($rx, $ry);

}

// ------------------------------------------------

// Lineare Regression (X → Y)

// Liefert Steigung b, Achsenabschnitt a, R²

// ------------------------------------------------

function linearRegression(array $x, array $y): ?array {

$n = count($x);

if ($n !== count($y) || $n === 0) return null;

$meanX = array\_sum($x) / $n;

$meanY = array\_sum($y) / $n;

$num = 0.0; $den = 0.0;

for ($i = 0; $i < $n; $i++) {

$num += ($x[$i] - $meanX) \* ($y[$i] - $meanY);

$den += ($x[$i] - $meanX) \*\* 2;

}

if ($den == 0) return null;

$b = $num / $den; // Steigung

$a = $meanY - $b \* $meanX; // Achsenabschnitt

// Bestimmtheitsmaß R²

$predY = [];

for ($i = 0; $i < $n; $i++) {

$predY[] = $a + $b \* $x[$i];

}

$r2 = pearsonCorrelation($y, $predY) \*\* 2;

return ['a' => $a, 'b' => $b, 'r2' => $r2];

}

// ------------------------------------------------

// Werte aus DB holen

// ------------------------------------------------

$query = "SELECT l.teilnehmer\_id, l.lernfortschritt, d.methodenvielfalt, e.interesse, s.h\_bedeutung

FROM mtr\_leistung l

JOIN mtr\_didaktik d ON l.teilnehmer\_id = d.teilnehmer\_id

JOIN mtr\_emotions e ON l.teilnehmer\_id = e.teilnehmer\_id

JOIN frzk\_semantische\_dichte s ON l.teilnehmer\_id = s.teilnehmer\_id

ORDER BY l.teilnehmer\_id";

$stmt = $pdo->query($query);

$data = [];

foreach ($stmt as $row) {

$tid = $row['teilnehmer\_id'];

if (!isset($data[$tid])) {

$data[$tid] = ['x' => [], 'y' => [], 'z' => [], 'h' => []];

}

$data[$tid]['x'][] = (float)$row['lernfortschritt'];

$data[$tid]['y'][] = (float)$row['methodenvielfalt'];

$data[$tid]['z'][] = (float)$row['interesse'];

$data[$tid]['h'][] = (float)$row['h\_bedeutung'];

}

// ------------------------------------------------

// Berechnungen für jeden Teilnehmer

// ------------------------------------------------

foreach ($data as $tid => $vals) {

$pearson = pearsonCorrelation($vals['x'], $vals['h']);

$spearman = spearmanCorrelation($vals['x'], $vals['h']);

$reg = linearRegression($vals['x'], $vals['h']);

if ($pearson !== null && $spearman !== null && $reg !== null) {

$insert = $pdo->prepare("

INSERT INTO frzk\_interdependenz

(zeitpunkt, x\_kognition, y\_sozial, z\_affektiv, h\_bedeutung, korrelationsscore, methode)

VALUES (NOW(), :x, :y, :z, :h, :score, :meth)

");

$insert->execute([

':x' => array\_sum($vals['x']) / count($vals['x']),

':y' => array\_sum($vals['y']) / count($vals['y']),

':z' => array\_sum($vals['z']) / count($vals['z']),

':h' => array\_sum($vals['h']) / count($vals['h']),

':score' => $pearson, // du kannst hier auch Spearman oder R² speichern

':meth' => "Pearson=" . round($pearson, 3) .

", Spearman=" . round($spearman, 3) .

", Regr[b]=" . round($reg['b'], 3) .

", R2=" . round($reg['r2'], 3)

]);

}

}

echo "frzk\_interdependenz mit Pearson, Spearman und Regression befüllt!\n";

**📘 Ergebnis**

* In der Spalte korrelationsscore steht der Pearson-Koeffizient (–1 … +1).
* In der Spalte methode steht eine **Kombination** aus:
  + Pearson
  + Spearman
  + Regression (Steigung b + R²).
* Damit hast du ein **Mehrmethoden-Ergebnis** je Teilnehmer.

👉 Soll ich dir zusätzlich noch ein **Aggregat über alle Teilnehmer** bauen (z. B. Gesamtkorrelationen für die ganze Gruppe statt pro Person)?

**Du:**

ja

**ChatGPT:**

Sehr gut 👍  
Dann erweitere ich dein Skript so, dass du **zwei Ebenen** bekommst:

1. **Individuelle Interdependenzen** (pro Teilnehmer → wie bisher)
2. **Aggregierte Interdependenzen** (über alle Teilnehmer → Gesamtwerte für die Gruppe)

**📄 frzk\_import.php (mit Aggregation)**

<?php

// ================================================

// FRZK: Interdependenzen berechnen (Einzel + Aggregat)

// ================================================

$dsn = "mysql:host=localhost;dbname=icas;charset=utf8mb4";

$user = "root";

$pass = "";

try {

$pdo = new PDO($dsn, $user, $pass, [

PDO::ATTR\_ERRMODE => PDO::ERRMODE\_EXCEPTION

]);

} catch (PDOException $e) {

die("DB-Verbindung fehlgeschlagen: " . $e->getMessage());

}

// ------------------------------------------------

// Pearson-Korrelation

// ------------------------------------------------

function pearsonCorrelation(array $x, array $y): ?float {

$n = count($x);

if ($n !== count($y) || $n === 0) return null;

$meanX = array\_sum($x) / $n;

$meanY = array\_sum($y) / $n;

$num = 0.0; $denX = 0.0; $denY = 0.0;

for ($i = 0; $i < $n; $i++) {

$dx = $x[$i] - $meanX;

$dy = $y[$i] - $meanY;

$num += $dx \* $dy;

$denX += $dx \* $dx;

$denY += $dy \* $dy;

}

if ($denX == 0 || $denY == 0) return null;

return $num / sqrt($denX \* $denY);

}

// ------------------------------------------------

// Spearman-Korrelation

// ------------------------------------------------

function spearmanCorrelation(array $x, array $y): ?float {

$n = count($x);

if ($n !== count($y) || $n === 0) return null;

$rank = function($arr) {

$sorted = $arr;

asort($sorted);

$ranks = [];

$rank = 1;

foreach ($sorted as $key => $val) {

$ranks[$key] = $rank++;

}

return $ranks;

};

$rankX = $rank($x);

$rankY = $rank($y);

return pearsonCorrelation(array\_values($rankX), array\_values($rankY));

}

// ------------------------------------------------

// Lineare Regression (X → Y)

// ------------------------------------------------

function linearRegression(array $x, array $y): ?array {

$n = count($x);

if ($n !== count($y) || $n === 0) return null;

$meanX = array\_sum($x) / $n;

$meanY = array\_sum($y) / $n;

$num = 0.0; $den = 0.0;

for ($i = 0; $i < $n; $i++) {

$num += ($x[$i] - $meanX) \* ($y[$i] - $meanY);

$den += ($x[$i] - $meanX) \*\* 2;

}

if ($den == 0) return null;

$b = $num / $den;

$a = $meanY - $b \* $meanX;

$predY = [];

for ($i = 0; $i < $n; $i++) {

$predY[] = $a + $b \* $x[$i];

}

$r2 = pearsonCorrelation($y, $predY) \*\* 2;

return ['a' => $a, 'b' => $b, 'r2' => $r2];

}

// ------------------------------------------------

// Werte aus DB holen

// ------------------------------------------------

$query = "SELECT l.teilnehmer\_id, l.lernfortschritt, d.methodenvielfalt, e.interesse, s.h\_bedeutung

FROM mtr\_leistung l

JOIN mtr\_didaktik d ON l.teilnehmer\_id = d.teilnehmer\_id

JOIN mtr\_emotions e ON l.teilnehmer\_id = e.teilnehmer\_id

JOIN frzk\_semantische\_dichte s ON l.teilnehmer\_id = s.teilnehmer\_id

ORDER BY l.teilnehmer\_id";

$stmt = $pdo->query($query);

$data = [];

foreach ($stmt as $row) {

$tid = $row['teilnehmer\_id'];

if (!isset($data[$tid])) {

$data[$tid] = ['x' => [], 'y' => [], 'z' => [], 'h' => []];

}

$data[$tid]['x'][] = (float)$row['lernfortschritt'];

$data[$tid]['y'][] = (float)$row['methodenvielfalt'];

$data[$tid]['z'][] = (float)$row['interesse'];

$data[$tid]['h'][] = (float)$row['h\_bedeutung'];

}

// ------------------------------------------------

// 1. Individuelle Interdependenzen speichern

// ------------------------------------------------

foreach ($data as $tid => $vals) {

$pearson = pearsonCorrelation($vals['x'], $vals['h']);

$spearman = spearmanCorrelation($vals['x'], $vals['h']);

$reg = linearRegression($vals['x'], $vals['h']);

if ($pearson !== null && $spearman !== null && $reg !== null) {

$insert = $pdo->prepare("

INSERT INTO frzk\_interdependenz

(zeitpunkt, x\_kognition, y\_sozial, z\_affektiv, h\_bedeutung, korrelationsscore, methode)

VALUES (NOW(), :x, :y, :z, :h, :score, :meth)

");

$insert->execute([

':x' => array\_sum($vals['x']) / count($vals['x']),

':y' => array\_sum($vals['y']) / count($vals['y']),

':z' => array\_sum($vals['z']) / count($vals['z']),

':h' => array\_sum($vals['h']) / count($vals['h']),

':score' => $pearson,

':meth' => "Individuell | Pearson=" . round($pearson, 3) .

", Spearman=" . round($spearman, 3) .

", Regr[b]=" . round($reg['b'], 3) .

", R2=" . round($reg['r2'], 3)

]);

}

}

// ------------------------------------------------

// 2. Aggregierte Interdependenzen (Gruppe)

// ------------------------------------------------

$allX = []; $allY = []; $allZ = []; $allH = [];

foreach ($data as $vals) {

$allX = array\_merge($allX, $vals['x']);

$allY = array\_merge($allY, $vals['y']);

$allZ = array\_merge($allZ, $vals['z']);

$allH = array\_merge($allH, $vals['h']);

}

$pearsonGroup = pearsonCorrelation($allX, $allH);

$spearmanGroup = spearmanCorrelation($allX, $allH);

$regGroup = linearRegression($allX, $allH);

if ($pearsonGroup !== null && $spearmanGroup !== null && $regGroup !== null) {

$insert = $pdo->prepare("

INSERT INTO frzk\_interdependenz

(zeitpunkt, x\_kognition, y\_sozial, z\_affektiv, h\_bedeutung, korrelationsscore, methode)

VALUES (NOW(), :x, :y, :z, :h, :score, :meth)

");

$insert->execute([

':x' => array\_sum($allX) / count($allX),

':y' => array\_sum($allY) / count($allY),

':z' => array\_sum($allZ) / count($allZ),

':h' => array\_sum($allH) / count($allH),

':score' => $pearsonGroup,

':meth' => "Aggregiert | Pearson=" . round($pearsonGroup, 3) .

", Spearman=" . round($spearmanGroup, 3) .

", Regr[b]=" . round($regGroup['b'], 3) .

", R2=" . round($regGroup['r2'], 3)

]);

}

echo "Interdependenzen (individuell + Gruppe) erfolgreich berechnet!\n";

**📘 Ergebnis**

* **Pro Teilnehmer**: Ein Datensatz mit Pearson/Spearman/Regression (wie bisher).
* **Zusätzlich ein Gruppendatensatz**:
  + Mittelwerte aller Dimensionen X, Y, Z, H über alle Teilnehmer.
  + Ein Gesamtkorrelationsmaß für die gesamte Feldstudie.

bitte ein skript zur befüllung von frzk\_semantische\_dichte

**ChatGPT:**

Alles klar 👍  
Ich baue dir jetzt ein **PHP-Skript mit PDO**, das die Tabelle **frzk\_semantische\_dichte** automatisch befüllt.  
Die Idee: wir berechnen **semantische Dichte σ(t)** aus den bestehenden Daten (z. B. mtr\_leistung, mtr\_emotions, mtr\_didaktik) und speichern zusätzlich die **Ableitung dσ/dt** sowie einen **Stabilitätswert**.

**📄 frzk\_semantische\_dichte\_import.php**

<?php

// ================================================

// FRZK: Befüllung der Tabelle frzk\_semantische\_dichte

// ================================================

$dsn = "mysql:host=localhost;dbname=icas;charset=utf8mb4";

$user = "root";

$pass = "";

try {

$pdo = new PDO($dsn, $user, $pass, [

PDO::ATTR\_ERRMODE => PDO::ERRMODE\_EXCEPTION

]);

} catch (PDOException $e) {

die("DB-Verbindung fehlgeschlagen: " . $e->getMessage());

}

// ------------------------------------------------

// 1. Datenbasis holen

// ------------------------------------------------

// Annahme: σ entsteht aus Kombination von kognitiven, sozialen und affektiven Faktoren.

// Wir nutzen: lernfortschritt (kognitiv), methodenvielfalt (sozial), interesse (affektiv)

$query = "SELECT l.teilnehmer\_id, l.datum, l.lernfortschritt, d.methodenvielfalt, e.interesse

FROM mtr\_leistung l

JOIN mtr\_didaktik d ON l.teilnehmer\_id = d.teilnehmer\_id

JOIN mtr\_emotions e ON l.teilnehmer\_id = e.teilnehmer\_id

ORDER BY l.teilnehmer\_id, l.datum";

$stmt = $pdo->query($query);

// ------------------------------------------------

// 2. Werte pro Teilnehmer berechnen

// ------------------------------------------------

$data = [];

foreach ($stmt as $row) {

$tid = $row['teilnehmer\_id'];

$zeit = $row['datum'];

$x = (float)$row['lernfortschritt']; // Kognition

$y = (float)$row['methodenvielfalt']; // Sozial

$z = (float)$row['interesse']; // Affektiv

if (!isset($data[$tid])) {

$data[$tid] = [];

}

// σ als gewichtetes Mittel (Gewichte anpassbar)

$sigma = 0.5 \* $x + 0.3 \* $y + 0.2 \* $z;

$data[$tid][] = [

'zeitpunkt' => $zeit,

'sigma' => $sigma

];

}

// ------------------------------------------------

// 3. dσ/dt (Ableitung) und Stabilität berechnen

// ------------------------------------------------

foreach ($data as $tid => $werte) {

$prev = null;

foreach ($werte as $idx => $point) {

$zeit = $point['zeitpunkt'];

$sigma = $point['sigma'];

// Ableitung berechnen (Differenz zur vorherigen Messung)

if ($prev !== null) {

$deltaT = strtotime($zeit) - strtotime($prev['zeitpunkt']);

$dhdt = ($deltaT > 0) ? ($sigma - $prev['sigma']) / $deltaT : 0.0;

} else {

$dhdt = 0.0;

}

// Stabilität: inverse Varianz im 3er-Fenster

$window = array\_slice(

array\_column($werte, 'sigma'),

max(0, $idx - 2),

3

);

$mean = array\_sum($window) / count($window);

$var = 0.0;

foreach ($window as $val) {

$var += ($val - $mean) \*\* 2;

}

$stab = (count($window) > 1) ? 1 / (1 + $var) : 1.0;

// ------------------------------------------------

// 4. In DB schreiben

// ------------------------------------------------

$insert = $pdo->prepare("

INSERT INTO frzk\_semantische\_dichte

(teilnehmer\_id, zeitpunkt, h\_bedeutung, dh\_dt, stabilitaet\_score)

VALUES (:tid, :zeit, :sigma, :dhdt, :stab)

");

$insert->execute([

':tid' => $tid,

':zeit' => $zeit,

':sigma'=> $sigma,

':dhdt' => $dhdt,

':stab' => $stab

]);

$prev = $point;

}

}

echo "frzk\_semantische\_dichte erfolgreich befüllt!\n";

**📘 Erklärung**

* **σ (semantische Dichte)** = gewichtete Kombination von Lernfortschritt, Methodenvielfalt und Interesse.
  + Gewichtung (0.5 / 0.3 / 0.2) kannst du empirisch anpassen.
* **dσ/dt** = Änderungsrate pro Zeiteinheit (zeigt Lernsprünge oder Stagnation).
* **Stabilität** = Maß für Konstanz in einem kleinen Zeitfenster (inverse Varianz).
* Die Ergebnisse werden direkt in frzk\_semantische\_dichte gespeichert.