

CHECKLIST DE QA DIARIO PARA DATOS DE FUERZA

10 validaciones críticas antes de estimar un 1RM o ajustar la carga del microciclo. *Por Lisandro Cacciatore | Consultor en Datos Deportivos*

La Regla de Oro

La mayoría de los softwares y planillas de fuerza te calculan el tonelaje total y el 1RM estimado (e1RM) automáticamente. El problema es que asumen que el dato que ingresó el atleta o el entrenador es perfecto.

Un error de tipeo en el peso o un falso registro de velocidad en el encoder te rompen la tendencia de todo el mes. El software no tiene criterio; tu staff sí. Este checklist de 5 minutos blinda tu base de datos contra el error humano.

FASE 1: Estructura y Contexto (Antes de la barra)

- [] **1. Peso Corporal Actualizado:** ¿El atleta actualizó su peso corporal esta semana? Fundamental si vas a reportar métricas de fuerza relativa (Wilks, DOTS, o multiplicadores de peso corporal como "2x BW").
- [] **2. Readiness / Wellness Registrado:** Si cruzás carga externa (kilos) con interna (estado de recuperación), revisá que el cuestionario pre-sesión esté completo *antes* del primer levantamiento, no 8 horas después.
- [] **3. Nomenclatura Estricta:** ¿Es "Sentadilla Tras Nuca", "Back Squat" o "SQ"? Unificar la nomenclatura es el paso cero. Si el sistema o el Excel lee tres nombres distintos por error de tipeo, te genera tres historiales incompletos.

FASE 2: Carga Externa (Los Kilos y la Física)

- [] **4. Detección de "Typos" (Errores de tipeo):** Filtrá cargas que superen el 110% del 1RM histórico del atleta. Tipear 1500 kg en lugar de 150 kg es el error más común y rompe cualquier gráfico de volumen semanal.
- [] **5. Lógica de Repeticiones vs. Intensidad:** ¿El registro tiene sentido biológico? Si alguien registró 12 repeticiones con el 95% de su 1RM, hay dos opciones: el 1RM está desactualizado hace años, o mintió en el registro.

- [] **6. Limpieza de Encoders (Velocidades Fantasma):** Si medís VBT (m/s), buscá picos irreales mayores a 1.5 m/s en ejercicios de fuerza máxima. Generalmente es el impacto de la barra contra el rack al sacarla o guardarla. Ese dato debe borrarse.
- [] **7. Consistencia del Peso de la Barra:** ¿Se estandarizó si la carga total incluye la barra? Un error clásico en juveniles o planteles femeninos es usar barras de 15 kg y anotar asumiendo que pesa 20 kg.

FASE 3: Validación Subjetiva y de Estimación (El Esfuerzo)

- [] **8. Mismatch RPE/RIR vs. Velocidad:** Si el encoder marca 0.3 m/s (cerca del fallo) pero el atleta reporta un RPE 6 (esfuerzo medio), hay un problema de percepción o un error de registro. Esa serie no es válida para generar proyecciones.
- [] **9. Control de Fórmulas de e1RM:** Las fórmulas lineales (como Epley o Brzycki) se rompen a altas repeticiones. Asegurate de filtrar las series de hipertrofia (más de 8-10 reps) antes de que el sistema calcule el 1RM estimado diario.
- [] **10. Series de Calentamiento vs. Efectivas:** ¿Tu sistema suma los 60 kg de entrada en calor al "Volumen Total" de la sesión de fuerza máxima? Definí un umbral estricto (ej: solo sumar tonelaje por encima del 65% del 1RM o RPE > 6) para no ensuciar la métrica de volumen.

BONUS: Automatización con Python (Cheat Sheet)

No busques estos errores deslizando el dedo en Excel. Acá tenés un script básico usando la librería **pandas** que audita las anomalías más comunes de tu archivo CSV en un segundo.

Podés copiar y adaptar este código en tu entorno:

Python

```
None
import pandas as pd

# 1. Cargar el registro de la sesión (CSV exportado)
df = pd.read_csv('ironlog_sesion.csv')

# 2. Alerta de Errores de Típo (Cargas > 300kg que probablemente sean falsas)
typo_alert = df[(df['Peso_Kg'] > 300) & (df['Ejercicio'] != 'Leg Press')]
print("⚠️ ALERTA: Posibles errores de típo en Carga:")
```

```
print(typo_alert[['Atleta', 'Ejercicio', 'Peso_Kg']]))

# 3. Alerta de Lógica RPE vs RIR
# (Ej: Reportar RPE 10 pero anotar que quedaron repeticiones en
reserva)
rpe_logic_error = df[(df['RPE'] >= 9.5) & (df['RIR'] > 0)]
print("\n⚠ ALERTA: Inconsistencia en esfuerzo percibido:")
print(rpe_logic_error[['Atleta', 'Ejercicio', 'RPE', 'RIR']]))

# 4. Alerta de Velocidades Fantasma (VBT)
vbt_error = df[(df['Velocidad_ms'] > 1.5) &
(df['Ejercicio'].isin(['Squat', 'Bench', 'Deadlift']))]
print("\n⚠ ALERTA: Velocidades pico irreales (Revisar choques
con el rack):")
print(vbt_error[['Atleta', 'Ejercicio', 'Velocidad_ms']]))
```

Casos Reales (El Problema en la Práctica)

El Récord Mundial Accidental: Un atleta anotó "1505" en lugar de "150.5" en su Deadlift desde el celular. El dashboard del club le calculó un tonelaje semanal nivel mutante y destrozó la métrica de "Carga Crónica" de todo el equipo por un solo cero. Nadie lo notó hasta 15 días después.

El Falso Estancamiento: Un equipo reportaba estancamiento en la fuerza de sus atletas. Al auditar los datos, vimos que el sistema calculaba el e1RM usando todas las series. En pretemporada (series de 12 reps), la fórmula sobreestimaba los máximos. En temporada (bajaban a 3 reps), la fórmula era más precisa, pero el número daba más bajo. Hacía parecer que perdían fuerza. El problema era matemático, no biológico.

¿Tu equipo toma decisiones con información confiable?

Ordenar el proceso de captura y validación de datos es el único camino para que la información realmente impacte en el rendimiento.

Si tu institución necesita estructurar su arquitectura de datos deportivos, automatizar flujos de trabajo o auditar cómo están usando la información hoy, hablemos.

[\[Agendar una Revisión Inicial Sin Cargo \]](#)

Lisandro Cacciatore

lisandrocacciatore.com