Plano de Gestão de Dados

Nov 25, 2026 · @Heitor Santana

1. Identificação do projeto

Este plano define como os dados do projeto foram obtidos, organizados, verificados, armazenados, publicados e preservados, e fixou essas decisões antes da etapa de análise.

Item	Descrição
Título	Detecção de falhas no processo Tennessee Eastman por meio de resíduos de modelos NARX/NARMAX identificados a partir de dados abertos
Disciplina	PCS5787 Ciência dos Dados e Big Data, Programa de Pós-Graduação em Engenharia Elétrica, Escola Politécnica da USP, 3º quadrimestre de 2026
Equipe	Heitor Gonçalves Santana (Nº USP 17981494), Helena Bigliani Ferreira (Nº USP 12552502) e Gabriel Rios de Souza (Nº USP 18260136)
Objetivo	Verificar se modelos dinâmicos identificados apenas com dados de operação normal detectam e isolam falhas no TEP, percorrendo todas as etapas do ciclo de vida dos dados do DataONE
Modelo de referência	Ciclo de vida do DataONE (Michener et al., 2012) e princípios FAIR (Wilkinson et al., 2016)
Versão do plano	1.0

2. Descrição dos dados

O projeto reutiliza um conjunto aberto de simulações do TEP e produz dados derivados de menor volume, que são os únicos publicados pelo grupo.

Dados de origem. Utiliza-se o conjunto *Additional Tennessee Eastman Process Simulation Data for Anomaly Detection Evaluation* (Rieth et al., 2017), publicado no Harvard Dataverse com DOI 10.7910/DVN/6C3JR1. Empregam-se apenas os dois arquivos de treino, a saber: `TEP_FaultFree_Training.RData`, com 500 simulações normais, e `TEP_Faulty_Training.RData`, com 500 simulações para cada uma das 20 falhas. Cada

simulação tem 500 amostras (25 h, período de 3 min) e 55 colunas: `faultNumber`, `simulationRun`, `sample` e as 52 variáveis de processo (41 medidas e 11 manipuladas). Os arquivos somam 5,25 milhões de registros e ocupam cerca de 2,2 GB em memória em dupla precisão.

Dados derivados. São produzidos pelo código do projeto:

Conjunto	Conteúdo	Volume aproximado
Extratos	Simulações selecionadas, em precisão simples	215 MB
Banco de documentos	2 000 documentos: 500 simulações normais e as simulações 426 a 500 de cada falha, com o dicionário de dados, os modelos e as métricas	210 MB
Modelos	Estrutura, parâmetros e ERR dos modelos ARX, NARX, NARMAX e NARX neural dos cinco subsistemas	< 1 MB
Resíduos	Resíduos um passo à frente e de simulação livre das simulações de validação e teste, por condição	120 MB
Resultados	Limiares, taxas de detecção, falsos alarmes, atrasos, intervalos de confiança e isolamento	< 1 MB
Código e documentação	Scripts Python e versão MATLAB, dicionário de dados, README e metadados	< 5 MB

Os arquivos de teste do mesmo conjunto, com 960 amostras por simulação, não foram usados nesta versão do projeto.

3. Coleta, organização e documentação

Os dados são obtidos uma única vez do repositório de origem e, a partir daí, organizados por simulação, que é a unidade de consulta, de análise e de separação dos conjuntos.

1. **Obtenção.** Os dois RData são baixados do Dataverse, e registram-se o DOI, a versão (V1) e a data do acesso.
2. **Leitura.** Um leitor próprio do formato RData (XDR compactado) lê as colunas uma a uma e as converte para precisão simples, o que permite processar o arquivo de falhas num computador pessoal sem depender do R.

3. **Organização.** Cada simulação vira um documento com seus metadados (`faultNumber`, `simulationRun`, conjunto, DOI de origem, amostra de início da falha) e a matriz 500 × 52 como bloco binário.
4. **Separação.** A divisão é feita por simulação, e nunca por amostra, pois amostras de uma mesma simulação são correlacionadas no tempo: simulações normais 1 a 350 para identificação, 351 a 425 para validação e calibração dos limiares, e 426 a 500 para teste, normais e com falha.

Documentação. Mantêm-se os seguintes registros, a saber:

- dicionário de dados com nome, descrição, unidade, período de amostragem, média e desvio padrão de treino e indicação de uso de cada uma das 52 variáveis;
- README com proveniência, organização das pastas e instruções de reprodução;
- registros de execução da extração, com contagens de simulações e amostras;
- metadados de citação no esquema DataCite (`.zenodo.json`) e no formato CITATION.cff, com a relação *isDerivedFrom* apontando para o DOI de origem.

4. Formatos e padrões

Para publicação, preferem-se formatos abertos e legíveis sem software proprietário, e os formatos de trabalho ficam restritos à cópia local do grupo.

Dados	Formato de trabalho	Formato publicado	Justificativa
Dados de origem	RData (XDR, gzip)	Não publicados; referenciados pelo DOI	Formato da fonte
Extratos	NumPy `.npz` (Python) e `.mat` (MATLAB), precisão simples	Não publicados	Reprodutíveis a partir da fonte
Banco de documentos	Mongita, com a API do MongoDB, um documento por simulação	Não publicado	Reconstruído pelo código
Modelos e resultados	JSON	JSON	Estrutura hierárquica, texto aberto
Resíduos	NumPy `.npz`	Apache Parquet, por condição	Colunar, compacto, lido por Python, R e MATLAB

Dados	Formato de trabalho	Formato publicado	Justificativa
Métricas e dicionário	CSV	CSV (UTF-8)	Tabular, universal
Figuras e tabelas	PDF, PNG e LaTeX	Incluídas no artigo	Vetoriais para impressão
Código	Python 3 e MATLAB	`.py` e `.m`	Texto aberto, licença MIT
Metadados	JSON e YAML	`.zenodo.json` (DataCite) e CITATION.cff	Padrões do Zenodo e do GitHub

Os nomes de arquivo seguem um padrão único, `tep_<conteúdo>_<condição>_<conjunto>`, por exemplo `tep_residuos_falha19_teste.parquet`, e as variáveis mantêm os nomes da fonte (`xmeas_1` a `xmeas_41`, `xmv_1` a `xmv_11`).

5. Armazenamento, cópias de segurança e versionamento

Durante o projeto, os dados ficam numa pasta única, `PCS5787_Projeto_Final`, no computador do coordenador, com cópias segundo a regra 3-2-1: três cópias, em dois meios, uma delas fora do local de trabalho.

Cópia	Local	Conteúdo	Frequência
Principal	Computador do coordenador	Tudo: fonte, extratos, banco, derivados, código, artigo	Contínua
Segurança	Armazenamento em nuvem institucional da USP	Código, derivados, artigo e apresentação	Ao fim de cada etapa do ciclo de vida
Publicação	Zenodo	Pacote de depósito versionado	A cada versão publicada

Os dados de origem não entram nas cópias de segurança, pois permanecem disponíveis no Dataverse pelo DOI, e os extratos e o banco também não, pois são reconstruídos pelo código em menos de uma hora.

Versionamento. O pacote publicado recebe número de versão semântico (v1.0.0 na primeira publicação), registrado no README, no CITATION.cff e nos metadados; correções geram uma nova versão no Zenodo, que mantém o DOI de cada versão e um DOI conceitual comum a todas. Versões intermediárias do artigo são guardadas com sufixo numérico (`_v1`, `_v2`, ...), e nenhum arquivo de resultado é editado manualmente: todos são regenerados pelos *scripts*.

Segurança. Os dados são simulações de um processo de referência e não contêm informação pessoal, sigilosa ou industrial real, de modo que não exigem controle de acesso além do usual.

6. Garantia de qualidade

A qualidade é verificada em três níveis, a saber: na leitura dos dados, na análise exploratória e na reprodução dos resultados por implementações independentes.

Nível	Verificação	Critério de aceitação
Leitura	Cabeçalho, número e tipo das 55 colunas	Formato XDR e 55 colunas, ou a leitura é interrompida
Leitura	Contagem de simulações por falha e de amostras por simulação	500 simulações por condição e 500 amostras cada
Leitura	Valores ausentes e não finitos	Nenhum
Exploratória	Faixas, períodos de amostragem e correlações das variáveis	Exclusão justificada das 19 composições, amostradas a cada 6 ou 15 min
Separação	Ausência de vazamento entre treino, validação e teste	Conjuntos disjuntos por `simulationRun`
Modelos	Testes de correlação de Billings e R^2 em validação e teste	Diferença de R^2 entre validação e teste $\leqslant 0{,}01$
Reprodução	FROLS próprio comparado com a biblioteca SysIdentPy	Mesmos termos, mesma ordem, parâmetros iguais até a 6ª casa
Reprodução	Pipeline completo reimplementado em MATLAB e executado no GNU Octave	Modelos polinomiais e taxas de detecção sem rede neural idênticos

As tabelas do artigo são escritas diretamente pelos *scripts* a partir dos resultados, o que elimina erros de transcrição, e os resultados que dependem de números aleatórios (rede neural e *bootstrap*) usam semente fixa.

7. Aspectos legais, éticos e licenças

Os dados de origem são públicos e não são redistribuídos; os dados derivados e o código do grupo são publicados sob licenças abertas que exigem atribuição.

- **Dados de origem.** O uso segue os termos indicados na página do conjunto no Harvard Dataverse. O grupo não redistribui os arquivos RData: o pacote publicado os referencia pelo DOI, e o artigo e os metadados os citam como fonte (Rieth et al., 2017).
- **Dados derivados.** Licença Creative Commons Atribuição 4.0 Internacional (CC BY 4.0), que permite o reuso, inclusive comercial, desde que se cite a obra.
- **Código.** Licença MIT, com o texto da licença incluído no pacote.
- **Ética e privacidade.** Os dados são simulações de um processo químico de referência (Downs e Vogel, 1993), sem seres humanos, dados pessoais ou informação industrial sigilosa, e não requerem aprovação de comitê de ética nem tratamento sob a LGPD.
- **Autoria.** Os três integrantes figuram como autores do depósito, na ordem do artigo, com afiliação à Escola Politécnica da USP.

Pendência: confirmar na página do Dataverse a licença exata do conjunto de origem e registrá-la no README do pacote.

8. Compartilhamento e publicação

O pacote de dados derivados, código e documentação é depositado no Zenodo, na comunidade *Big Data Analytics Research Group of Escola Politécnica da Universidade de São Paulo*, onde recebe um DOI e fica acessível sem restrição.

Conteúdo do depósito.

- `dados/`: resíduos por condição (Parquet), modelos e resultados de detecção (JSON), métricas por falha e detector (CSV), dicionário de dados com os parâmetros de padronização e o relatório da verificação com a SysIdentPy;
- `codigo/`: *scripts* Python de todas as etapas e a versão MATLAB, com instruções de uso;
- artigo em PDF, apresentação e este plano de gestão de dados;
- README com proveniência e instruções de reprodução, licença do código, `.zenodo.json` e CITATION.cff.

Princípios FAIR.

Princípio	Como é atendido
Encontrável	DOI do Zenodo, metadados DataCite com título, autores, palavras-chave e comunidade

Princípio	Como é atendido
Acessível	Acesso aberto pelo protocolo HTTPS, sem cadastro
Interoperável	Formatos abertos (Parquet, CSV, JSON), vocabulário das variáveis igual ao da fonte, relação *isDerivedFrom* com o DOI de origem
Reutilizável	Licenças CC BY 4.0 e MIT, dicionário de dados, README e código que regenera todos os resultados

O DOI obtido é citado no artigo, no README, no CITATION.cff e no slide de publicação da apresentação.

9. Preservação de longo prazo e reutilização

A preservação de longo prazo fica a cargo do Zenodo, mantido pelo CERN, que guarda os arquivos e os metadados pelo tempo de vida do repositório e garante que o DOI continue resolvendo.

- **O que se preserva:** o pacote publicado, suficiente para reproduzir todas as tabelas e figuras do artigo a partir dos dados de origem.
- **O que não se preserva:** os extratos, o banco de documentos e os arquivos intermediários, pois são regenerados pelo código, e os dados de origem, já preservados pelo Dataverse.
- **Dependências:** o README registra as versões de Python, MATLAB ou Octave e das bibliotecas usadas, e a versão MATLAB dispensa *toolboxes*, o que reduz o risco de obsolescência do ambiente.
- **Reutilização prevista:** comparação de novos detectores com os mesmos resíduos e métricas, uso dos resíduos como assinatura num classificador de diagnóstico e aplicação do mesmo método aos arquivos de teste do conjunto de origem.

Pendência: incluir no README as versões de Python e das bibliotecas antes do depósito.

10. Responsabilidades e recursos

A gestão dos dados é coordenada por Helena Bigliani Ferreira, e cada etapa do ciclo de vida tem um responsável, conforme a divisão de trabalho descrita no artigo; a coordenação geral do projeto é de Heitor Gonçalves Santana.

Etapa (DataONE)	Ação	Artefato	Responsável
Planejar	Elaboração deste plano	Plano de gestão de dados	Helena Bigliani Ferreira
Coletar	Download dos RData e extração	Extratos e registros de execução	Helena Bigliani Ferreira
Assegurar	Verificações de leitura e análise exploratória	Relatório da EDA	Gabriel Rios de Souza
Descrever	Dicionário de dados e metadados	Dicionário, README, metadados	Helena Bigliani Ferreira
Preservar	Banco de documentos e cópias de segurança	Banco e cópias	Helena Bigliani Ferreira
Integrar	Esquema único por simulação	Coleções do banco	Helena Bigliani Ferreira
Analisar	Identificação dos modelos e detector por resíduos	Modelos, resíduos, métricas	Heitor Gonçalves Santana
Analisar	Linha de base PCA e avaliação estatística	Métricas e intervalos de confiança	Gabriel Rios de Souza
Publicar	Depósito no Zenodo e registro do DOI	Pacote publicado	Helena Bigliani Ferreira

O artigo e a apresentação são redigidos e revisados pelos três integrantes, e cada um confere o pacote antes do depósito.

Recursos. O projeto não tem custo: usa computador pessoal, software livre (Python, GNU Octave, Mongita), armazenamento institucional da USP e o Zenodo, gratuito para depósitos até 50 GB.

Referências

- DOWNS, J. J.; VOGEL, E. F. A plant-wide industrial process control problem. *Computers & Chemical Engineering*, v. 17, n. 3, p. 245-255, 1993.
- MICHENER, W. K. et al. Participatory design of DataONE: enabling cyberinfrastructure for the biological and environmental sciences. *Ecological Informatics*, v. 11, p. 3-15, 2012.

- RIETH, C. A.; AMSEL, B. D.; TRAN, R.; COOK, M. B. *Additional Tennessee Eastman Process Simulation Data for Anomaly Detection Evaluation*. Harvard Dataverse, V1, 2017. DOI: 10.7910/DVN/6C3JR1.
- WILKINSON, M. D. et al. The FAIR guiding principles for scientific data management and stewardship. *Scientific Data*, v. 3, art. 160018, 2016.